NONLINEAR PREDICTIVE CONTROL USING PARTICLE SWARM OPTIMIZATION: APPLICATION TO POWER SYSTEMS

BY

MUHAMMAD SALMAN YOUSUF

A Thesis Presented to the

DEANSHIP OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

In

ELECTRICAL ENGINEERING

JUNE 2009

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
DHAHRAN 31261, SAUDI ARABIA

DEANSHIP OF GRADUATE STUDIES

This thesis, written by **MUHAMMAD SALMAN YOUSUF** under the direction of his thesis adviser and approved by his thesis committee, has been presented to and accepted by the Dean of Graduate Studies, in partial fulfillment of the requirements for the degree of **MASTER OF SCIENCE IN ELECTRICAL ENGINEERING**.

Thesis Committee

H. Duwaish
Dr. Hussain N. Al-Duwaish (Adviser)

Dr. Zakariya M. Al-Hamouz (Co-advisor)

Dr. Jamil M. Bakhashwain (Member)

Dr. Samir A. Al-Baiyat (Member)

Dr. Ahmed A. Masoud (Member)

20 JAN 2010
Dr. Samir H. Abdul-Jauwad
Department Chairman

Dr. Salam A. Zummo
Dean of Graduate Studies

30/1/10
Date

To My Family - I can never thank you enough for your infinite love and relentless support.

To My Teachers - Each of you has contributed to shape my world as it is today.

To My Friends - May your worlds be filled with the excitement of discovery.

ACKNOWLEDGMENTS

I begin with the name of Allaah, the Most Merciful, the Most Beneficent

All praise is due to Allaah, the source of all knowledge, blessings and strength. I acknowledge His infinite mercy and grace in making this work a success. And may Allaah send His eternal peace and continuous blessings upon his final messenger Muhammad, who is the inspiration of our lives, and upon his family and his companions until the Day when we shall all meet Him. I am indebted to various persons and organizations for their contribution towards the successful completion of this thesis and I would like to express my gratitude to them here.

I am extremely grateful to King Fahd University of Petroleum & Minerals for giving me the opportunity to carry out this work under the guidance of a scholarly faculty, generous research resources, and a stimulating work environment.

My sincere gratitude and thanks go to my thesis advisor Dr. Hussain N. Al-Duwaish and co-adviser Dr. Zakariya M. Al-Hamouz. I am thankful to them for their meticulous attention, useful guidance and patience. They have given me huge favors and generous amount of time and learning, for which I am eternally grateful. I would also like to extend my appreciation to my thesis committee members, Dr. Jamil M. Bakhashwain, Dr. Samir A. Al-Baiyat and Dr. Ahmed A. Masoud for their

encouragement and cooperation. I would like to especially thank Dr. Masoud for advising me on my thesis presentation and helping to make it more effective.

I would like to thank my friend Muhammad Shakil for his discussions on several parts of my thesis, and for imparting some of the key concepts I have used. I am grateful to my friends Zeeshan, Ahmed and Saqib. This work would have certainly been a lot more difficult, if they wouldn't have shared the hardwork, frustrations and joys it brought. Their friendship, support and encouragement certainly brought happiness and created countless precious memories which are now inedible. Special thanks to the rest of my friends, Babar, Asif, Mazhar Bhai, Bilal Bhai and Akhlaq Bhai for bringing happiness and support.

I would also like to thank my colleagues and brothers at the Procter & Gamble Company, Sarmad Ahmed, Sharafaldin Hasan and Anas Almounir for sharing some of my burdens. I would like to especially thank my boss, Daoud Abulnabi for giving me the support and flexibility to work on the final leg of my thesis.

I would like to acknowledge the pivotal role of NED University of Engineering & Technology, Karachi, for giving me the essential foundation of the concepts related to engineering, as well as life, during my Bachelors degree there.

My deepest gratitude goes out to my resilient and loving family of continuous high achievers. My respect and gratitude towards my parents cannot be expressed in words. They have been my first school and the reason of whatever faculties I am able to display today. I would like to acknowledge my brother Hassaan and my sisters Sameen and Sireen for being the source of amazing support, strength and inspiration,

all at the same time. Lastly, my deep sense of gratitude to Nazwar Kanwal, my wife, for her endless prayers, deep concern and strong support.

TABLE OF CONTENTS

LIST OF TABLES viii

LIST OF FIGURES ix

ABSTRACT (ENGLISH) xvi

ABSTRACT (ARABIC) xviii

NOMENCLATURE xix

1 INTRODUCTION 1

1.1 Motivation 1

1.2 The Model Predictive Control Theory and Background 5

1.3 The Particle Swarm Optimization Technique 16

1.3.1 Particle Swarm Optimization Algorithm 24

1.4 Statement of Problem 28

1.5 Thesis Organization and Scope 31

2 PSO BASED MODEL PREDICTIVE CONTROL: MPC-PSO 33

2.1 The Predictive Controller Concept 33

2.1.1 Summary of Model Predictive Control Algorithm 35

2.2 Elements of the Predictive Controller Design 36

2.2.1 Process Models 36

2.2.2 Cost Functions 39

2.2.3 Process Constraints 41

2.3 PSO based Model Predictive Control: MPC-PSO 44
2.3.1 Implementation of MPC-PSO 44
2.3.2 Constraint Formulation 49
2.4 MPC-PSO Explained through Nonlinear Control of a Valve 50
2.4.1 Regulatory Control of Control Valve Using MPC-PSO 54
2.4.2 Servo Control of Control Valve Using MPC-PSO 55
2.5 Advantages and Disadvantages of MPC 56
2.6 Conclusion 57

3 APPLICATION OF MPC-PSO TO SYNCHRONOUS MACHINE ON INFINITE BUS (SMIB) SYSTEM 59
3.1 Introduction 59
3.2 Nonlinear Model of the SMIB System 61
3.3 MPC-PSO Control of SMIB System 64
3.3.1 Performance of Proposed Controller for Perturbed System . . . 65
3.3.2 Performance of Proposed Controller Under Fault Conditions . . 74
3.4 Conclusion 80

4 APPLICATION OF MPC-PSO TO LOAD FREQUENCY CONTROL 82
4.1 Introduction 82
4.2 Model of an LFC System 88
4.3 Single Area Load Frequency Control 91
4.3.1 Single Area LFC Excluding Nonlinearity 92
4.3.2 LFC Including GRC Nonlinearity 97
4.3.3 LFC Including Nonlinearity and Parameter Variations 104
4.4 Two-Area Load Frequency Control 107
4.4.1 Two-Area LFC Excluding Nonlinearity 109
4.4.2 Two-Area LFC Including GRC Nonlinearity 114
4.5 Four-Area Load Frequency Control 117
4.6 Conclusion 122

5 APPLICATION OF MPC-PSO TO A FOSSIL FUEL POWER UNIT **125**

5.1 Introduction 125

5.2 Model of Drum-Boiler-Turbine Power Unit 128

5.3 Operating Points of Drum-Boiler-Turbine Power Unit 131

5.4 Problem Formulation 131

5.5 Operation at Nominal Operating Points 132

5.6 Control of Boiler-Turbine System for Switching OPs 134

5.7 Conclusion 139

6 CONCLUSIONS AND RECOMMENDATIONS **144**

6.1 Further Topics 144

6.1.1 MPC-PSO in Noisy Environment 144

6.1.2 MPC-PSO with Model Mismatch 147

6.1.3 Effect of Prediction Horizon 149

6.1.4 Effect of PSO Population Size 150

6.1.5 Computational Complexity and Real-Time Application 151

6.2 Summary 153

6.3 Conclusion 155

6.4 Recommendation for Further Work 157

REFERENCES **160**

VITAE **184**

LIST OF TABLES

5.1 Operating Points for the System 131

6.1 Modeling Errors for SMIB Parameters 147

6.2 Effects of Modeling Errors on System Parameters 147

LIST OF FIGURES

1.1 Structure of MPC 8

1.2 Predicted output and the corresponding optimal input over a horizon H_p, where *u(k)* is the optimal input, $\widehat{y}(k)$ is the predicted output and *y(k)* is the process output. 9

1.3 Flowchart of a simple Particle Swarm Optimization Algorithm 21

1.4 Structure of Proposed MPC-PSO Controller 29

1.5 Power System Dynamics Tested with the Proposed Controller 32

2.1 Structure of Proposed MPC-PSO Controller 45

2.2 Flowchart of Proposed PSO-based MPC 46

2.3 Nonlinear Characteristic of a Control Valve 52

2.4 Set Point Tracking of Control Valve for Proposed Controller 55

2.5 Control Effort for Set Point Tracking of Control Valve 55

2.6 Servo Behavior of Control Valve for Proposed Controller 55

2.7 Control Effort for Servo Behavior of Control Valve 55

3.1 Diagram of Synchronous Machine Infinite Bus (SMIB) Power System . 61

3.2 Angle (rad) for Case I 68

3.3 Convertor Current 68

3.4 Frequency Deviation (rad/sec) 68

3.5 Mechanical Power 68

3.6 Control Input 1, $\cos(\beta)$ 68

3.7 Control Input 2, v 68

3.8 Angle for $|u| \leq \pm 1.5$ 70

3.9 Convertor Current 70
3.10 Frequency Deviation (rad/s) 70
3.11 Mechanical Power 70
3.12 Control Input 1, cos(β) 70
3.13 Control Input 2, v 70
3.14 Angle (rad) for Case II 73
3.15 Convertor Current 73
3.16 Frequency Deviation (rad/sec) 73
3.17 Mechanical Power 73
3.18 Control Input 1, cos(β) 73
3.19 Control Input 2, v 73
3.20 Angle (rad) for Case III 75
3.21 Convertor Current 75
3.22 Frequency Deviation (rad/sec) 75
3.23 Mechanical Power 75
3.24 Control Input 1, cos(β) 75
3.25 Control Input 2, v 75
3.26 Angle (rad) for Case IV 77
3.27 Convertor Current 77
3.28 Frequency Deviation (rad/sec) 77
3.29 Mechanical Power 77
3.30 Control Input 1, cos(β) 77
3.31 Control Input 2, v 77
3.32 Angle (rad) for Case V 79
3.33 Convertor Current 79
3.34 Frequency Deviation (rad/sec) 79
3.35 Mechanical Power 79
3.36 Control Input 1, cos(β) 79
3.37 Control Input 2, v 79

4.1 Block diagram of single area LFC 89
4.2 Disturbance and Frequency Deviation for Designs 1-3 94
4.3 Change in Generated Power for Designs 1-3 94
4.4 Control Effort for Designs 1-3 94
4.5 Cost Function for Designs 1-3 94
4.6 Robustness of Frequency Deviation 96
4.7 Robustness in Change in Generated Power 96
4.8 Control Efforts 96
4.9 Cost Functions 96
4.10 Block diagram of nth area LFC with GRC nonlinearities 98
4.11 Frequency Deviation for GRC = 0.01 100
4.12 Frequency Deviation for GRC = 0.0017 100
4.13 Generated Power Output for GRC = 0.01 100
4.14 Generated Power Output for GRC = 0.0017 100
4.15 Control Effort for GRC = 0.01 100
4.16 Control Effort for GRC = 0.0017 100
4.17 Cost for GRC = 0.01 101
4.18 Cost for GRC = 0.0017 101
4.19 Varying Disturbance Applied 102
4.20 Frequency Deviation for Varying Disturbance 102
4.21 Generated Power Output for Varying Disturbance 102
4.22 Control Effort for Varying Disturbance 102
4.23 Frequency Deviation for Varying GRC 103
4.24 Generated Power Output for Varying GRC 103
4.25 Control Effort for Varying GRC 103
4.26 Cost Functions for Varying GRC 103
4.27 Frequency Deviation for GRC with Parameter Variation 105
4.28 Generated Power Output for GRC with Parameter Variation 105
4.29 Control Effort for GRC with Parameter Variation 105
4.30 Control Effort for GRC with Parameter Variation 105

4.31 Block diagram of two-area LFC with GRC nonlinearities 107
4.32 Frequency Deviation in Area 1 110
4.33 Frequency Deviation in Area 2 110
4.34 Change in Generated Power in Area 1 110
4.35 Change in Generated Power in Area 2 110
4.36 Control Effort, u_1 110
4.37 Control Effort, u_2 110
4.38 Power Flow in Tie-Line 111
4.39 Cost Function 111
4.40 Frequency Deviation in Area 1 113
4.41 Frequency Deviation in Area 2 113
4.42 Change in Generated Power in Area 1 113
4.43 Change in Generated Power in Area 2 113
4.44 Control Effort, u_1 113
4.45 Control Effort, u_2 113
4.46 Power Flow in Tie-Line 114
4.47 Cost Function 114
4.48 Frequency Deviation in Area 1 116
4.49 Frequency Deviation in Area 2 116
4.50 Change in Generated Power in Area 1 116
4.51 Change in Generated Power in Area 2 116
4.52 Control Effort, u_1 116
4.53 Control Effort, u_2 116
4.54 Power Flow in Tie-Line 117
4.55 Block diagram of four-area interconnected power system 118
4.56 Area 1 of four-area interconnected power system 119
4.57 Four-Area Excluding Nonlinearity, Δf_1 & Δf_2 120
4.58 Δf_3 & Δf_4 120
4.59 ΔP_{g1} & ΔP_{g2} 120
4.60 ΔP_{g3} & ΔP_{g4} 120

4.61 u_1 & u_2 120
4.62 u_3 & u_4 120
4.63 Power Flow in Tie-Lines for Four-Area System Excluding GRC 121
4.64 Four-Area Including Nonlinearity, Δf_1 & Δf_2 123
4.65 Δf_3 & Δf_4 123
4.66 ΔP_{g1} & ΔP_{g2} 123
4.67 ΔP_{g3} & ΔP_{g4} 123
4.68 u_1 & u_2 123
4.69 u_3 & u_4 123
4.70 Power Flow in Tie-Lines for Four-Area System with GRC 124

5.1 Block diagram of Drum Boiler Turbine System 129
5.2 Power Generated at Operating Point 4 133
5.3 Pressure at Operating Point 4 133
5.4 Water Level at Operating Point 4 133
5.5 u_1 at Operating Point 4 133
5.6 u_2 at Operating Point 4 133
5.7 u_3 at Operating Point 4 133
5.8 Power Generated - Switch Operating Points from 4 to 5 135
5.9 Pressure - Switch Operating Points from 4 to 5 135
5.10 Water Level - Switch Operating Points from 4 to 5 135
5.11 u_1 - Switch Operating Points from 4 to 5 135
5.12 u_2 - Switch Operating Points from 4 to 5 135
5.13 u_3 - Switch Operating Points from 4 to 5 135
5.14 Power Generated - Switch Operating Points from 4 to 6 137
5.15 Pressure - Switch Operating Points from 4 to 6 137
5.16 Water Level - Switch Operating Points from 4 to 6 137
5.17 u_1 - Switch Operating Points from 4 to 6 137
5.18 u_2 - Switch Operating Points from 4 to 6 137
5.19 u_3 - Switch Operating Points from 4 to 6 137

5.20 Power Generated - Switch Operating Points from 4 to 7 140
5.21 Pressure - Switch Operating Points from 4 to 7 140
5.22 Water Level - Switch Operating Points from 4 to 7 140
5.23 u_1 - Switch Operating Points from 4 to 7 140
5.24 u_2 - Switch Operating Points from 4 to 7 140
5.25 u_3 - Switch Operating Points from 4 to 7 140
5.26 Power Generated - Switch Operating Points from 4 to 3 141
5.27 Pressure - Switch Operating Points from 4 to 3 141
5.28 Water Level - Switch Operating Points from 4 to 3 141
5.29 u_1 - Switch Operating Points from 4 to 3 141
5.30 u_2 - Switch Operating Points from 4 to 3 141
5.31 u_3 - Switch Operating Points from 4 to 3 141
5.32 Power Generated - Switch Operating Points from 4 to 2 142
5.33 Pressure - Switch Operating Points from 4 to 2 142
5.34 Water Level - Switch Operating Points from 4 to 2 142
5.35 u_1 - Switch Operating Points from 4 to 2 142
5.36 u_2 - Switch Operating Points from 4 to 2 142
5.37 u_3 - Switch Operating Points from 4 to 2 142

6.1 Block Diagram of Proposed MPC-PSO with Noise 145
6.2 Noise in the System 146
6.3 Converter Current for SMIB system with Noise 146
6.4 Frequency Deviation with Noise 146
6.5 Control Effort, $cos(\beta)$ for Noise 146
6.6 Control Effort, v for Noise 146
6.7 Converter Current for SMIB system with Model Mismatch 148
6.8 Frequency Deviation with Model Mismatch 148
6.9 Control Effort, $cos(\beta)$ for Model Mismatch 148
6.10 Control Effort, v for Model Mismatch 148
6.11 Effect of Prediction Horizon - Change in Frequency Deviation 149

6.12 Effect of Prediction Horizon - Change in Power Generated 149
6.13 Effect of Population Size - Change in Frequency Deviation 151
6.14 Effect of Population Size - Change in Power Generated 151

THESIS ABSTRACT

NAME:	Muhammad Salman Yousuf
TITLE OF STUDY:	Nonlinear Predictive Control using Particle Swarm Optimization: Applications to Power Systems
MAJOR FIELD:	Electrical Engineering
DATE OF DEGREE:	June 2009

Although classical control is still the workhorse in the majority of control engineering applications, it is well recognized that this linear control method is not always the optimum way to deal with the typical highly nonlinear plants. This is especially true for multivariable systems with a variety of constraints. Due to recent and continual increment in the complexity of systems and tighter product specifications, the quality requirements from automatic control have increased greatly. At the same time, the available computing power has rose to fantastic levels. Consequently, computationally intensive control methods can be applied to complex systems comparatively easily. Model Predictive Control (MPC) techniques were developed to obtain tighter control and they were applied successfully to several industrial applications. MPC re-

quires the solution of a constrained optimization problem at each sampling instant. This optimization is carried out by various techniques like linear programming (LP), quadratic programming (QP), dynamic programming (DP) and heuristics like Genetic Algorithms (GA).

In this thesis, a new implementation of MPC is proposed using Particle Swarm Optimization (PSO) algorithm. The proposed method formulates the MPC as an optimization problem and PSO is used to optimize it. This gives numerous advantages like adaptability, possibility of varying control objectives, and enhanced capability of handling constraints. The proposed method is applied to an area of engineering systems that has been relatively unexplored by MPC, i.e. power systems. Both SISO and MIMO nonlinear systems are considered. Three practical Power System problems are considered and the proposed technique is applied to them.

Keywords: *MPC, Model Predictive Control, Nonlinear Control, Particle Swarm Optimization, Prediction Horizon, Control of Power Systems, Load Frequency Control, Synchronous Machine on Infinite Bus, Fossil Fuel Power Unit, Boiler-Turbine System, Predictive Control of Power Systems*

خلاصــة الرســالــة

الاسم الكامل : محمد سلمان يوسف

عنــوان الرسالة : التحكم بالتنبؤ الغير خطي بواسطة تذبذب الجزيئات الأمثل - التطبيقات في أنظمة القوى الكهربائية

التخـصـص : هندسة الكهربائية

تاريخ الشهادة : يونيو 2009م

على الرغم من أن التحكم الكلاسيكي هو المحرك لأغلبية تطبيقات هندسة التحكم. إن من المعروف أنه نظرية التحكم الخطي ليست دائما الأفضل للتحامل مع الأنظمة الغير خطية العالية. هذا صحيح بالأخص للأنظمة متعددة المتغيرات مع التنوع للقيود. بسبب الزيادة المستمرة والحديثة في الأنظمة المعقدة وخواص المنتج، متطلبات الجودة للتحكم الأوتوماتيكي قد زادت بشكل كبير. في الوقت نفسه الطاقة المحوسبة المتواجدة قد ارتفعت بشكل رائع. تبعا لذلك ، نظريات التحكم المتزايدة للحوسبة قابلة للتطبيق لأنظمة معقدة بشكل سهل نسبيا. تقنيات (MPC) طورت للحصول على تحكم أفضل وقد طبقت بنجاح للتطبيقات الصناعية. ال(MPC) تتطلب الحلول للمشاكل المثلى مع القيود عند لحظة أخذ العينات. هذا التمثيل طبعة بواسطة تقنيات متعددة مثل البرمجة الخطية والبرمجة التربيعية والبرمجة الديناميكية والخوارزمية الجينية.

في هذه الرسالة تطبيعية جديد بـ(MPC) قد طرح باستعمال خوارزمية التذبذب الأمثل. النظرية المطروحة تشكل ال(MPC) على أنها مشكلة تحسينية وال(PSO) تستعمل لتحسينها. هذا يعطي ميزه هائلة مثل القدرة على التكيف واحتمالية تعريف أي هدف للتحكم ويحسن القدرة على معالجة القيود. النظرية المطروحة طبقت في مجال هندسة الأنظمة التي قد لم تكشف بواسطة ال(MPC) وهي أنظمة الطاقة الكهربائية. قد اعتبرت كل من الأنظمة غير الخطية (SISO) و (MIMO). ثلاثة مشاكل لأنظمة الطاقة اعتبرت والتقنيات المطروحة قد طبقت لهم.

درجة الماجستير في العلوم
جامعة الملك فهد للبترول والمعادن
يونيو 2009م

Nomenclature

SISO	:	Single Input Single Output
MIMO	:	Multi Input Multi Output
EA	:	Evolutionary Algorithm
PSO	:	Particle Swarm Optimization
MPC	:	Model Predictive Control
PID	:	Proportional Integral Derivative
MPHC	:	Model Predictive Heuristic Control
DMC	:	Dynamic Matrix Control
GPC	:	Generalize Predictive Control
EPSAC	:	Extended Prediction Self Adaptive Controller
PFC	:	Predictive Functional Controller
EHAC	:	Extended Horizon Adaptive Control
CSTR	:	Continuously Stirred Tank Reactor
LQ	:	Linear Quadrature
ARMA	:	Autoregressive Moving Average
MAC	:	Model Algorithmic Controller
EPSAC	:	Extended Prediction Self Adaptive Control
STEC	:	Spring Thermal Energy Conversion
EC	:	Evolutionary Computation
CPSO	:	Cooperative Particle Swarm Optimization

MO	:	Multi-objective
HPSO	:	Hybrid Particle Swarm Optimization
EPSO	:	Extended Particle Swarm Optimization
LFC	:	Load Frequency Control
AGC	:	Automatic Generation Control
NARMAX	:	Nonlinear Auto Regressive and Moving Average with Exogenous Inputs
TS	:	Tabu Search
VSC	:	Variable Structure Controller
DC	:	Direct Current
AC	:	Alternating Current
HVDC	:	High Voltage Direct Current
GRC	:	Generation Rate Constraint
SCC	:	State Contractive Constraint
SMIB	:	Synchronous Machine on Infinite Bus
FFPU	:	Fossil Fuel Power Unit

CHAPTER 1

INTRODUCTION

1.1 Motivation

The vast majority of industrial processes are typically operated using linear controllers, although it is well know that these processes are highly nonlinear. For these nonlinear systems, designing a feedback control law is very difficult because of the necessity to explore the whole state space. For nearly five decades, the PID controller has been the premium controller used in industries and other applications. It has shown good results, especially for single-input single-output applications. For multi-variable systems, it has shown limited success. The limitations of the PID controller are due to its characteristics, mainly its hardware realizability [41].

However, due to the continual evolution of digital computers, research to produce better control systems has increased and advanced control systems are being developed. These systems are digital and do not need hardware realizability. Also, the

processes today are much more complex and they need to be optimized and strictly controlled. This is especially true for multi-variable systems. Furthermore, rising costs of energy and raw materials, combined with the availability of powerful and economical microprocessors have provided additional incentive to evolve better control techniques.

Therefore, as we can now pursue the development of advanced control more feasibly than ever before, and the demands of our time are also increasingly stringent, we have no lack of motivation to develop and use better and advanced control methods. These motivating points can be enumerated as follows:

1. Limitations of classical (PID) control, especially for multivariable, nonlinear, complex systems.

2. Requirement of high steady state optimization and optimal closed loop control.

3. Strong emphasis on smooth plant operation with minimum down time.

4. Strong emphasis on optimal plant economics and efficiency.

5. Modern processes becoming ever-more complex.

6. Rising costs of energy, raw materials, labor etc.

7. Rising demands of consumers.

8. Realization that maximum profit can be generated by responding effectively to marketplace variations with minimum investment using integration of all aspects of automation.

9. Systems being forced to operate over wider range of conditions for maximum efficiency.

10. Availability of amazingly powerful computer and microprocessor technology at much lower costs.

So, being motivated by these factors, a modern day control problem that any control system must solve can be stated as follows [112].

Online updation of the manipulated variables to satisfy multiple, changing performance criteria in the face of changing plant characteristics.

Today, all control methodologies are dedicated to the solution of this problem. These methodologies are distinguishable by the technique they use, which lie in the distinction of one or many of the following entities:

1. Initial assumptions.

2. Mathematical formulation of performance criteria.

3. Selection of process representation.

4. Intended hardware implementation.

5. Intended process efficiency.

6. Importance of process constraints.

7. Process complexity.

8. Importance of uncertainties in the process.

9. Economic goals.

Many control methodologies are available today with different sub-techniques. The aim is to simply the system and the performance objectives mathematically such that the control problem defined by [112] is solved in the best possible and optimal way while fulfilling the hardware limitations. Perhaps the most crucial part is the incorporation of constraints in the problem formulation. We will discuss this point further in this thesis. The dynamic objective function is a mathematical function that has to be minimized. This together with the dynamic inequality constraints on the system formulate the performance criteria. The system is represented using the usual well-known methods as a dynamic model, with uncertainties, if present. Lately, these uncertainties are recognized as extremely important for optimal plant operation and are incorporated in the controller formulation.

At this point, the following point is worth emphasizing: The operating point that will satisfy all the economic goals of the process will lie at the intersection of the constraints. So, to be optimal, any control system must anticipate constraint violations and compensate for them in a systematic way. Violations must not be allowed while keeping the operating point as close as possible to these constraints.

Thus, the fundamentally important control problem is the creation of an effective, practical method for the design of feedback controllers for constrained dynamic linear or nonlinear systems with some uncertainties. Usually, designers using different control techniques ignore the issue of constraints at the design stage and deal with them in an ad hoc way during the implementation. However, Model Predictive Control

(MPC) technique provides a way and methodology to handle system constraints in a systematic way, both during design and implementation and to optimize the control. Furthermore, MPC technique is independent of the type of system model, objective functions, and constraints. Thus, currently, it is the only control methodology that can most directly incorporate all the process constraints, different performance criteria and is capable of utilizing any available process model [112]. This is why it is extremely successful in numerous applications.

To search for optimal control, MPC can be combined with many heuristic techniques, for example, Particle Swarm Optimization.

1.2 The Model Predictive Control Theory and Background

In this section, an introduction of the Model Predictive Control (MPC) theory and algorithm is given along with a substantial literature review.

The current interest of the industry in MPC can be traced back to a set of papers which appeared in the late 1970s. In 1978 Testud, Richalet, Rault, and Papon described successful applications of "Model Predictive Heuristic Control" [147] and in 1979 engineers from Shell, Cutler and Ramaker, outlined "Dynamic Matrix Control" (DMC) and reported applications to a fluid catalytic cracker [37]. Then in 1987 Mohtadi et al. presented the "Generalized Predictive Control" [97], [98]. In these techniques or algorithms, an explicit dynamic model of a plant is used to predict the

effect of future actions of the manipulated variables on the output, thus providing the name "Model Predictive Control". The future moves of the manipulated variables are determined by optimization with the objective of minimizing the predicted error subject to operating constraints. The optimization is repeated at each sampling time based on updated information i.e. measurements from the plant.

The success of MPC can be attributed to the following three factors:

1. Incorporation of an explicit process model in the control calculation allowing the controller to deal directly with all significant features of the plant dynamics.

2. Consideration of the plant behavior over a future horizon in time, allowing the effects of feedforward and feedback disturbances to be anticipated and removed in advance. This allows the controller to drive the plant more closely along the desired future trajectory.

3. Consideration of process input, state and output constraints directly in control calculations. Thus control violations are less likely to occur.

As mentioned earlier, it is this inclusion of constraints that most clearly distinguishes MPC from other process control paradigms [113]. Some good reviews of model predictive control can be found in [34], [120], [121], [114], [105], [88] and [117].

Besides DMC and MPHC. there are several other commercially available model predictive controllers available today, some of them are:

- MAC (Model Algorithmic Controller) or MPHC (Model Predictive Heuristic Controller)

- GPC (Generalized Predictive Controller)
- EPSAC (Extended Prediction Self Adaptive Controller)
- PFC (Predictive Functional Controller)
- EHAC (Extended Horizon Adaptive Control)

to name a few.

In model predictive control, the process output is predicted by using a model of the process to be controlled. Any model that describes the relationship between the input and the output of the process can be used. Further, if the process is subjected to disturbances a disturbance or noise model can be added to the process model thus allowing the effect of the disturbances on the predicted process output to be taken into account. In order to define how well the predicted process output tracks the reference trajectory, a criterion function is used. Typically the criterion is the difference between the predicted process output and the desired reference trajectory. A simple criterion function [115] is:

$$J = \sum_{i=1}^{N} [\widehat{y}(k+i) - w(k+i)]^2 \tag{1.1}$$

where $\widehat{y}$ is the predicted process output, w is the reference trajectory, and H_p is the predicted horizon. The basic structure common to all MPC algorithms is shown in Figure 1.1.

Here the controller output sequence u_{opt} over the prediction horizon is obtained by minimization of J with respect to u. As a result the future tracking error is minimized. If there is no model mismatch i.e., the model is identical to the process and there are

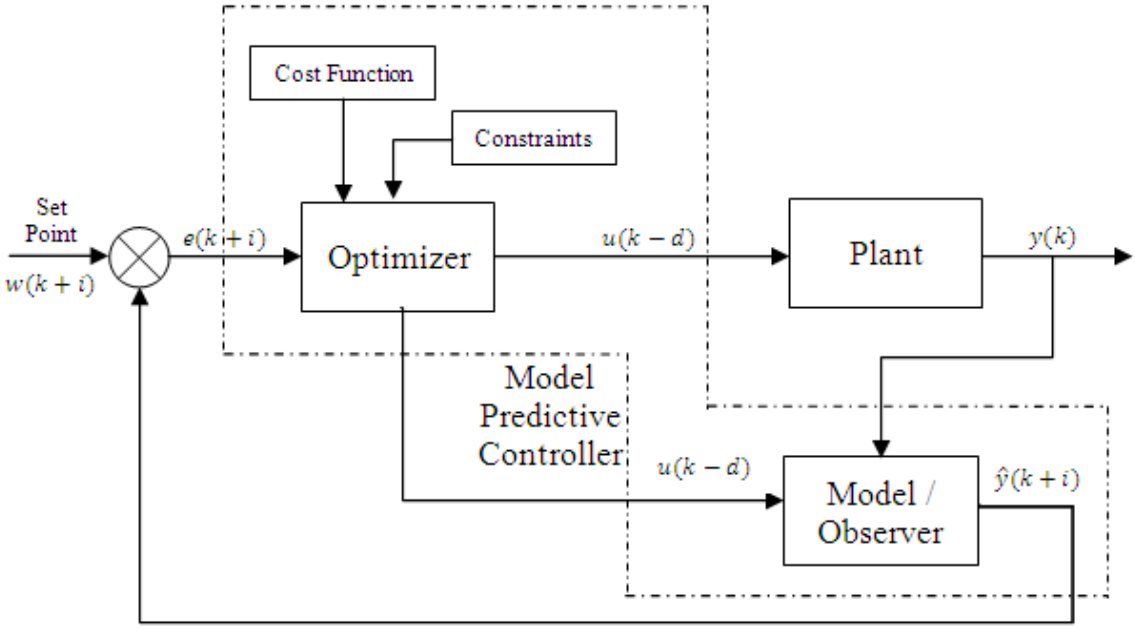

Figure 1.1: Structure of MPC

no disturbances and constraints, the process will track the reference trajectory exactly on the sampling instants.

The key to a good MPC algorithm is the technique used to optimize the cost function, J. In this thesis, Particle Swarm Optimization is used for this purpose, the details of which will see in Section 1.3. Figure 1.2 shows the predicted output and input over a horizon.

The working of MPC is well-known in the literature and will be discussed in detail Section 2. However, this section will elaborate more on the applications and techniques of MPC as found in the extensive literature available on the topic. For example, in the sections 5 of the online publications [158], [159], and [160], the International Journal of Robust and Nonlinear Control has mentioned several papers related to nonlinear control and constraints. However, there are numerous other publications available.

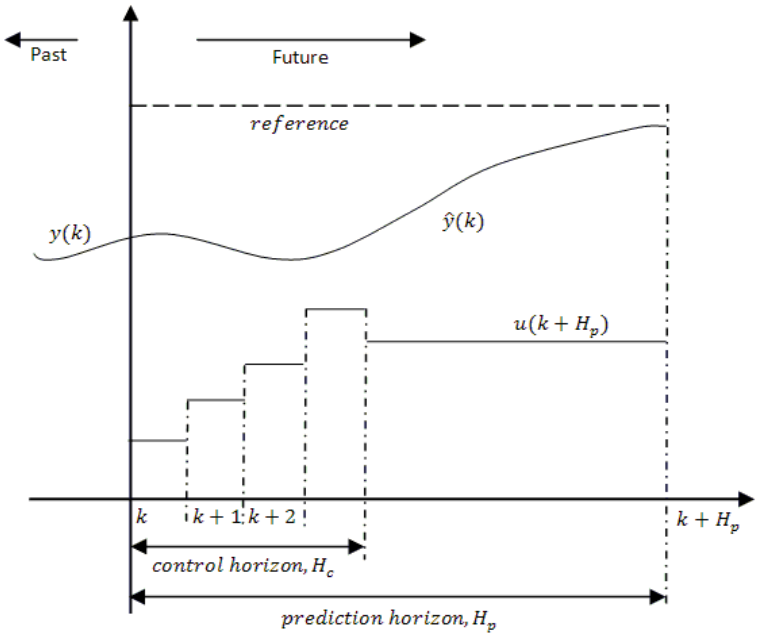

Figure 1.2: Predicted output and the corresponding optimal input over a horizon H_p, where *u(k)* is the optimal input, $\widehat{y}(k)$ is the predicted output and *y(k)* is the process output.

As mentioned already, the MPC technique is not new. Continuing from that, Yu-geng in 1993 demonstrated the effectiveness of MPC on an inverted pendulum on a cart using a two-level model predictive control algorithm [172]. At the first level, the nonlinear plant is controlled with a state-feedback system to linearize the closed-loop system. At the second level, the MPC algorithm is applied for further linearizing the system. This achieves good dynamics and robustness. This approach also decreases the computational complexity. Similarly, in 1994, the control of a paper machine headbox was achieved with a nonlinear predictive controller incorporating a nonlinear model-based observer within the MPC framework [127]. The proposed approach was shown to handle deterministic disturbances. constraints in the manipulated variables and mismatch between model and process. Under very restrictive assumptions the

algorithm was interpreted as a linearizing controller. In 1995 and 1996, the publications of Yim [169] [168], demonstrate an approach to end point trajectory control of elastic manipulators based on the nonlinear predictive control theory. Also the 1995 publication of Patwardhan [109], demonstrating the performance of the MPC algorithm by simulating a benchmark CSTR (Continuous Stirred Tank Reactor) control problem, and as the already referred [77], in which Katende demonstrates adaptive GPC algorithm on nonlinear models are notable. All of this research indicates that better results are achieved in the respective applications through the deployment of some form of MPC.

A method of transforming a general constrained nonlinear optimization problem into a convex optimization problem with quadratic criterion function and linear constraints, enabling the solution to be found through quadratic programming routines was given in [17] in 1996. It uses a neural network model incorporated in a feedback linearization scheme. With a suitable approximation of the resulting nonlinear constraints, it is then possible to solve the MPC with feedback linearization by also using a quadratic programming optimization routine. Simulation results show enhancement of the overall closed-loop performance with no penalizing effect on the control effort. In the author's opinion these are promising results in the neural control field of nonlinear processes subjected to environmental constraints.

In the literature, different MPC structures and slightly varying algorithms are abundantly found. The typical way to present these is to apply the algorithm to a challenging problem/application and observe the results. For example, Multivariable

MPC applied to cement mills showed improved performance compared to previous LQ techniques with respect to the hardness of the raw material as in [93]. Also, the application of cascaded MPC to an induction motor by Hedjar [60], [62] and [63] gave more effective control. When considering both electrical and mechanical dynamics of the motor, the proposed control law dealt with mismatched parameters as well as disturbance in the load effectively. In an interesting paper, Zhang and Morris have shown an approach to partition a process into several fuzzy operating regions which are identified as linear models of the parts of the whole process using neuro-fuzzy approach [173]. Based on these local linear models, a nonlinear MPC is developed by combining several local linear MPCs, which usually give analytical solutions. The technique is demonstrated for the pH control of a CSTR.

Nonlinear MIMO MPC was presented by Al-Duwaish (2003) and applied to various chemical processes including a shell distillation column by allowing the impurity and pressure control of the column to be decoupled while satisfying the constraints [7]. Rueda et al. also applied MPC to a high-purity distillation column in 2005, based on iterative linearization of the model response so that the same closed loop responses as in the pure non-linear approach are obtained but with reduced computation times and more efficient optimization [125]. MPC based on neural networks can was also applied to cascaded thermal processes by Pappa in 2004 [106]. Among the newer applications, a new constrained multivariable predictive control scheme is proposed to maximize the production in polymerization processes by Alamir et al., 2007 [5]. The key features of the proposed feedback strategy are its ability to rigorously handle the process con-

straints, for example, input saturation, maximum allowed heat production, maximum temperature values, and rate of change, as well as its real time implementability due to the low dimensional control parametrization being used. Application of an efficient MPC algorithm to a Hot Strip Finishing Mill can be found in [20].

MPC also finds applications in the field of robotics. An example is to control a manipulator with a flexible forearm [140], where a neural network represents the dynamical model. The nonlinear predictive controller was designed using the receding horizon approach. The aim was to design a controller using the neural network and also predict the output. This application was demonstrated by Song and Koivo in 1999 and is more related to neural networks, but it does use the receding horizon technique as a part. MPC is also applied for the nonlinear feedback control of robotic manipulator by Hedjar [61]. In 2005, Hedjar et al. also demonstrated a finite horizon nonlinear predictive controller with integral action and applied it to rigid link manipulators resulting in asymptotic tracking of smooth trajectories and robustness to parameter uncertainties [64]. A nonlinear low dimensional predictive control approach is applied for the control of RABBIT, a walking under-actuated robot. The resulting technique seems to be real-time implementable thanks to the low dimension of the optimization problem [30] (2005). MPC is also used to control the level of blood glucose in Type I diabetic patients, as recently demonstrated in [71]. Practical tests showed robustness and advantages of using nonlinear MPC. [44] discusses the application of MPC to a 2-degree of freedom helicopter model. The algorithm keeps the helicopter stable in maneuvers which were not possible with a linear controller.

To check the working of an MPC algorithm, instead of using actual systems, non-linear models such as the Hammerstein and Wiener or a combination of both can be used. Such models are well-known in the literature. Examples of such efforts can be found in [16] and [53].

As regards to the variety of algorithms and design of an MPC controllers, again the literature available is abundant. For example, the actual design of an MPC using 9 neural networks is given in [90] by Liu. A neural network based nonlinear predictive controller can also be found in Lazar 2001 [87] and a recent approach through this area is in [72], where Support Vector Machine Neural Network is used to devise a methodology for the generalized predictive control scheme. Similarly the first steps in obtaining the theory to build a Fuzzy based MPC was published in 1999 by Espinosa [48]. Another Fuzzy based approach is given in [29]. A fairly new technique for MPC is by using Multirate Nonlinear Predictive Control as proposed by Halldorson [58], [57] . This new approach reduces the computational load because it divides the prediction horizon into only a few equidistant intervals with piecewise constant control signals. After solving a first dynamic optimization problem the prediction horizon is halved, keeping the second half of the solution fixed and doubling the sampling rate in the first half of the control horizon. Using these settings a second optimization is performed to improve the first acquired solution. This procedure is repeated until the applied control step has a reasonable sampling time. This improved stability and computational cost properties of the resulting control strategy without deteriorated quality of control. MPC schemes can also be based on nonlinear state space models

as shown by Rau in 2002 [116].

Multivariable processes using MPC as shown to be more identifiable in closed loop with techniques based on autoregression models (ARMA), state space models, and neural networks [2].

The technique that is most closely related to the work proposed here than others, is the application of MPC with Genetic Algorithms to Nonlinear Models by Al-Duwaish and Naeem [2]. This is a big leap forward because it was the first time an Evolutionary Algorithm (EA, discussed in Section 1.3) was applied to MPC. The application involved use of the proposed MPC-GA on Hammerstein and Wiener Models. The novel application formulated the MPC as an optimization problem and and the genetic algorithm is used in the optimization process. This was presented as an MS Thesis by Naeem and Al-Duwaish in 2001 [100]. The GA-based MPC discussed in the work was demonstrated successfully to SISO, MIMO, linear, and nonlinear systems. This technique gave excellent control of a nonlinear control valve and a heat exchanger. Simulation results on a binary distillation column and are demonstrated successfully also. It is demonstrated that MPC can also be applied to power systems, namely a STATCOM single machine power system. This was the first time MPC was applied to any power system.

Several other practical and well researched MPC algorithms can be found in Zheng 1998 [175], Kouvaritakis 1999 [84], Cannon et al. 2001 [23], Imsland 2004 [70].

A valuable paper focused on the study of robustness properties of a family of MPC controllers is by Magni (2002) [92]. It solves the tracking problem for asymptotically

constant reference signals with emphasis given to integral action added to the plants. Advancements to the techniques related to the MPC algorithm include observer-based MPC [122], a fast modular multivariable nonlinear MPC [146], Nonlinear MPC with Extended Dynamic Matrix Control [49], and other techniques as in [9], [11], [110], [82], and [18]. The MPC concept can be extended to more advanced approaches based around it, such as Extended Prediction Self Adaptive Control (EPSAC) and Nonlinear-EPSAC [144] and can be applied to real-life systems.

Classical and very useful knowledge base for MPC is available in [113], [19], [77], [103], [102], and [96].

Recent additions to MPC literature are applying MPC to nonlinear processes [10] and discussion of nonlinear MPC for realtime applications [12]. Recently, Xu presented a GPC-PID control strategy for a cooling-coil unit in heating, ventilation and air conditioning systems. Results showed that the proposed controller was able to deal with a wide range of operating conditions and achieved better performance than conventional methods [164]. Predictive control of a complex district heating network is given in [126]. Other applications of MPC available in the literature are the control of power converters [36] based on the cost function, multi-objective nonlinear predictive control for hot Spring Thermal Energy Conversion (STEC) plant in Japan [174], verifying feasibility and effectiveness, a guidance law based on nonlinear predictive control for unmanned attack air vehicle [52], and pendulation control of an offshore crane [142]. An application involving improvement of the distribution system stability of gas turbines is given in [75]. MPC is used to successfully damp the oscillation

when the power distribution system is subjected to a disturbance. A low computation and simple structure predictive controller for MIMO HVAC systems is given in [163]. Song proposed a chaotic PSO-based Neural Network Predictive Controller for nonlinear systems in [141]. A robust model predictive control method is developed for the level control problem of a nuclear steam water generator by Hu and Yuan in [68]. Closed loop stability and constraints satisfaction in the entire operating range are guaranteed by the feasibility of the optimization problem and simulation results show the effectiveness of the proposed method.

1.3 The Particle Swarm Optimization Technique

Particle swarm optimization (PSO), first introduced by James Kennedy and Russel Eberhart, is one of the modern heuristic algorithms which belongs to the category of *swarm intelligence* methods [80]. It was developed through simulation of a simplified social system and was first presented by them in an IEEE Neural Network Conference paper [81]. In the simulation model of a social system, each particle position can be thought of being a state of mind of a particular setting of abstract variables that represent the individual's beliefs and attitudes. The movement of the particles thus correspond to the change of these concepts. Swarms or social groups adjust their beliefs and attitudes through the evaluation of stimuli from the environment and compare it to their existing knowledge. If such stimuli or values are found to be more fit, they replace their existing values. These three important properties of human or animal social behavior i.e., evaluation, comparison, and imitation, are the inspiration

for the particle swarm optimization algorithm, and the particle swarm uses these concepts in adapting to the environmental changes and solving complex minimization problems [81].

Besides being a model of human or animal behavior, the particle swarm is closely related to swarm intelligence. Here, there is no central control and no one gives orders. Each particle is a simple agent acting upon local information. But the swarm, as a whole, is able to perform tasks whose degree of complexity is well beyond the capabilities of an individual particle. This is due to self-organization. The interactions among the particles (low-level components) result in complex structures at the swarm (high-level or global) level making is possible for it to perform optimization of complex functions.

Kennedy and Eberhart later developed this algorithm and laid the foundation to what today is called as *Swarm Intelligence* [80], where they defined the five basic principles on which Swarm Intelligence is based upon. These five basic principles are:

1. The proximity principle: The swarm should be able to carry out simple time and space calculations.

2. The quality principle: The swarm should be able to respond to quality factors in the environment.

3. The principle of diverse response: The swarm should not commit its activities along excessively narrow channels.

4. The principle of stability: The swarm should not change its behavior every time the environment changes.

5. The principle of adaptability: The swarm must be able to change its behavioral mode when its worth the computational price.

Note that the fourth and fifth principles are the opposite sides of the same coin. The PSO algorithm adheres to all of these principles [81]. Thus, the PSO system is thought of as an intelligent system. This is because it is based upon artificial life and has roots in Evolutionary Computation (EC).

Like in other Evolutionary Computation methods, the particle swarm consists of a population of individuals that represent solutions to the optimization problems we need to be solved. An optimal solution is selected through an iterative and probabilistic modification of these solutions. There is not much difference in PSO and other Evolutionary Algorithms (EAs) in EC-terms. However, the difference lies in how the population/swarm is changed from one iteration to the next. In EAs, genetic operators like selection, mutation and crossover are used whereas in PSO, the particles are modified according to two formulas after each iteration. Also, conceptually, in PSO, the particles stay *alive* and inhibit the search space during the whole run, where as in other EAs, the individuals are replaced in each generation. Another fundamental conceptual difference is that in EA the objective is reach through *competitive search* whereas in PSO, it is reached through *cooperative search*.

As a result, PSO differs from other EAs [50] [56] [118] [136] in terms of performance. The EA techniques have been successfully applied in many areas. However, PSO is a more robust and fast algorithm that can provide better solutions to nonlinear, non-differentiable, multi-modal problems. Such problems involve the minimization of a

static objective function i.e., the main goal of a global minimizer that does not change. A high-quality solution can be generated within shorter calculation time with more stable convergence characteristic than given by other stochastic methods. Due to this ability, it is effective in solving problems in a wide variety of scientific fields [108].

Since it was conceived, several variants of PSO have surfaced. The drawback of the simple PSO was discussed by Angeline [8] and Kennedy [78]. The V_{max} operator introduced by Kennedy and Eberhart [81] is what makes the PSO algorithm a competitor of other EA algorithms as recognized by Angeline [8]. The most commonly used PSO algorithm that is used is given in [133] which is briefly discussed here.

As in other EAs, a population of individuals exist in PSO. However, here instead of using genetic operators, these individuals are "evolved" by cooperation and competition among themselves through generations. Each particle adjusts its "flying" according to its own experience as well as its companions' experience. Each individual, called a "particle" in fact, represents a potential solution to the problem.

Each particle is treated as a point in D-dimensional space. The ith particle is represented as

$$X_i = (x_{i1}, x_{i2}, ..., x_{iD}) \tag{1.2}$$

The best previous position (the position giving the best fitness value) of any particle is recorded and represented as

$$P_i = (p_{i1}, p_{i2}, ..., p_{iD}) \tag{1.3}$$

Similarly, the position change (velocity) of each particle is

$$V_i = (v_{i1}, v_{i2}, ..., v_{iD}) \tag{1.4}$$

The particles are manipulated according to the following equations which give the updated velocities and positions of the particles.

$$V_i^{n+1} = w * V_i^n + c_1 * r_{i1}^n * (P_i^n - X_i^n) + c_2 * r_{i2}^n * (P_g^n - X_i^n) \tag{1.5}$$

$$X_i^{n+1} = X_i^n + x * V_i^{n+1} \tag{1.6}$$

For easier understanding, a simple flowchart illustrating the general working of the PSO algorithm is given on the next page in Figure 1.3.

Later versions of the PSO discuss the effects of inertial weights which effect the change in velocity. The velocity is multiplied by an inertia factor before updating [135]. A variant is the accommodation of a constriction factor given by Clerc in [35]. This constricts the velocity of the particles to their present velocity before updating.

PSO converges very rapidly for uni-modal problems, but for multi-modal problems there is a risk of the algorithm getting stuck in the local optima, i.e. premature convergence. In order to avoid this, one would have to look into all possible local optima before deciding on the global optimal value. This by itself is cumbersome because the algorithm will take a large amount of time to journey through all the local solutions before converging on the global optima. One method to avoid this premature convergence was addressed in [91] wherein the researchers renewed the

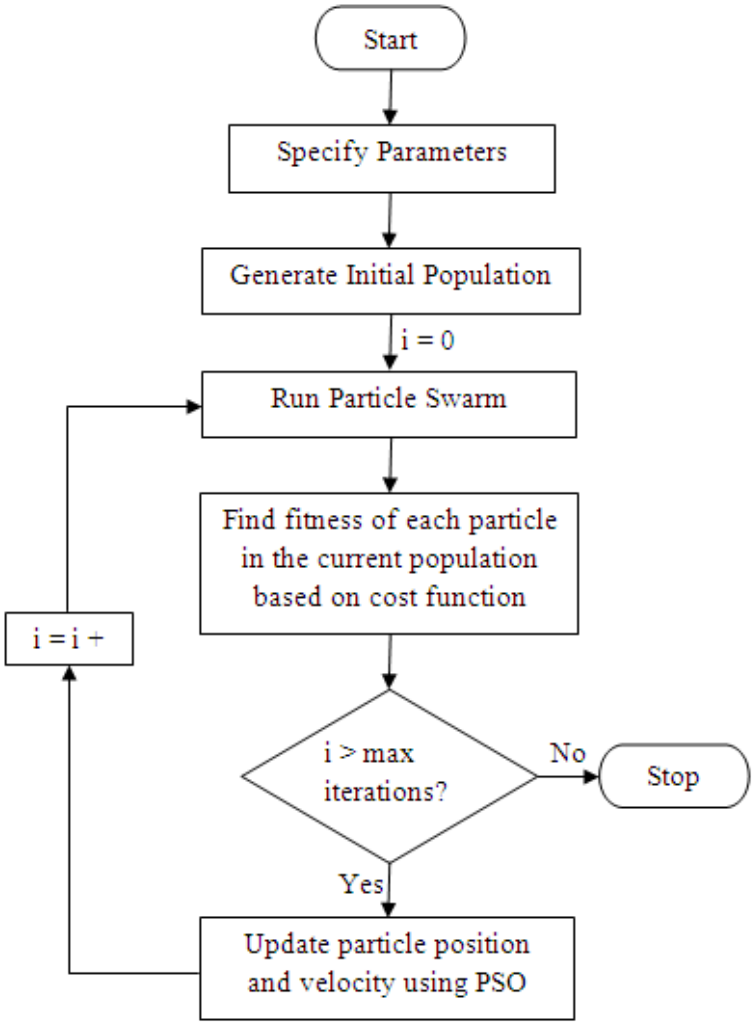

Figure 1.3: Flowchart of a simple Particle Swarm Optimization Algorithm

swarm by breeding some of the particles and called this algorithm the Hybrid-PSO, which is a merger of EAs with the basic PSO. Shi, et al. have also addressed this idea in a 2003 paper which proposes two new methods to integrate the PSO and Genetic Algorithm (GA) methods in parallel and in series. Simulations show that both the algorithms work better to obtain global minimum than standard PSO algorithm on a series of benchmark functions [132].

One of the most important factors for an efficient PSO algorithm is the proper

tuning of the parameters. This will enable the achievement of the global optimum quickly and without premature convergence. Numerous papers have discussed this matter and studied the effects of the tuning and parameter values. The prominent papers are Shi and Eberhart [134] and [135], Eberhart and Shi [45], Carlisle and Dozier [25], Angeline [8], and El-Gallad [47]. However, there are many others.

The idea of neighborhoods for PSO was discussed by Kennedy in [78]. Here the information about the solution can be shared between particles so as to enhance the search space and achieve better convergence. The particles can share information with two or more of their adjacent particles. Different types of neighborhood techniques which can influence the convergence were also discussed.

PSO was first applied to the training of Neural Networks [81] and since then it is very important in that regard. A development to the PSO algorithm in this regard is the Cooperative Particle Swarm Optimizer (CPSO), that leads to a significant reduction in the training time [150], [152]. The CPSO technique splits the solution vector of the function into several smaller vectors, where each sub-vector is optimized using a separate PSO. The effects of swarm size on CPSO are discussed further in [151]. A first study of PSO for multi-objective (MO) problems is presented by Parsopoulos and Vrahatis in 2002 [107]. The PSO technique solved well-known test problems, including difficult MO cases efficiently which is also compared with GA techniques.

Another advantage of PSO is that it can be used as a hybrid with different other optimization algorithms, for example, GA [132]. Another hybrid algorithm in PSO is based on Levenberg-Marquardt optimizer. This method gave a very high success rate

for six different parameter estimation problems [76]. Wang proposed a hybrid PSO algorithm with Cauchy mutation in 2007 [154]. This avoids the PSO from falling into a local optima as the best solution because the particles quickly get closer to the best particle. By adding a Cauchy mutation on the best particle, the mutated best particle leads other particles to better optima. Experimental study showed that this HPSO worked successfully with difficult multimodal functions. Another technique to avoid the local optima is using a dimension mutation operator. It is used in conjunction with a dynamically changing inertial weight based on the degree of particle diversity and improvement in fitness. This gives very good results as demonstrated experimentally by Wei in [156] by the use of several benchmark functions.

An Extended PSO (EPSO) algorithm was proposed by Xu in 2005. This algorithm uses local as well as global best positions for the calculation of the particles velocity at each iteration [162]. Thus the new equation for velocity becomes:

$$V_i^{n+1} = K.(V_i^n + c_1 * r_{i1}^n * (P_i^n - X_i^n) + c_2 * r_{i2}^n * (P_l^n - X_i^n) + c_3 * r_{i3}^n * (P_g^n - X_i^n)) \quad (1.7)$$

where K is the constriction factor given by Clerc [35]. This algorithm combines the advantages of global best and local best together. However, this needs further investigation in terms of weight assignments (c's), topology for local best, and further comparisons.

Eberhart and Shi have presented a very useful paper on the developments, applications and resources of the PSO in 2001 which can be found at [46]. This paper also list the main applications where PSO excels. PSO can be used to optimize a wide array

of functions of different types and is continually applied to a huge number of problems and systems. To make sure that while using the algorithm, the good practices are adopted and the bad practices are ignored, Kennedy's recent paper is very important [79]. It gives an informal discussion of the algorithm and its different parameters and emphasizes that the real research goal is not to make the algorithm more complicated. In fact, the goal is to strip it down to its essentials, at least while this paradigm is still young, and avoid suboptimal methods.

1.3.1 Particle Swarm Optimization Algorithm

A brief description of the PSO algorithm is described in Section 1.3 and a flowchart is given in Figure 1.3. Here how to implement the algorithm is briefly discussed and some of its attributed at the algorithmic level are given.

This algorithm is for the optimization of continuous and real-valued functions in the n-dimensional space, $\Re^n$. The PSO is a population-based search-algorithm and the population is called a swarm, S. The swarm consists of a number of particles that move around in the search space S. A neighborhood relation N is defined on the swarm. N determines the values of all particles, p_i and p_j, whether they are neighbors or not. Thus for each particle p a neighborhood, $N(p)$, containing all neighbors of p is assigned. A fitness function f must be defined to compare candidate solutions in the search space S which is a subset of $\Re^n$, and map into the real numbers, i.e.: $f : S \subseteq \Re^n \rightarrow \Re^n$. In fact, the PSO only compares fitness, so an ordinal fitness function would suffice. Each particle p has two state variables:

- Its current position: $\overrightarrow{x}(t)$
- Its current velocity: $\overrightarrow{v}(t)$

As well as a small memory containing:

- Its best position: $\overrightarrow{p}(t)$
- The best $\overrightarrow{p}(t)$ of all $p \in N(p)$:$\overrightarrow{g}(t)$,

Where $\overrightarrow{x}(t)$, $\overrightarrow{v}(t)$, $\overrightarrow{p}(t)$ and $\overrightarrow{g}(t)$ are n-dimensional vectors.

Particle Swarm Optimization basically consists of three parameters:

- v_{max} which restricts every coordinate of $\overrightarrow{v}(t)$ within the range [-v_{max} to v_{max}],
- ϕ_1 and ϕ_2 that determine the influence of $\overrightarrow{p}(t)$ and $\overrightarrow{g}(t)$ in the velocity update formula.

The swarm is initialized at time $t = 0$ by placing the particles randomly and uniformly distributed in S and assigning a random and uniformly chosen velocity vector $\overrightarrow{v}(0)$ from V^n. Moreover, set $\overrightarrow{p}(t) = \overrightarrow{g}(t) = \overrightarrow{x}(t)$.

The iterative optimization process starts after this initialization. The expressions for the particle positions and velocities in the next time step are given by these recursive equations:

$$\overrightarrow{v}(t+1) = \overrightarrow{v}(t) + \phi_1(\overrightarrow{p}(t) - \overrightarrow{x}(t)) + \phi_2(\overrightarrow{g}(t) - \overrightarrow{x}(t)) \quad (1.8)$$

$$\overrightarrow{x}(t+1) = \overrightarrow{x}(t) + \overrightarrow{v}(t+1) \quad (1.9)$$

Note that these equations are just another way to represent equations 1.5 and 1.6.

The position of a particle at time t+1 is calculated as a sum of it old position $\overrightarrow{x}(t)$ and current velocity $\overrightarrow{v}(t+1)$. Additionally, the velocity $\overrightarrow{v}(t+1)$ is updated as a sum of the particle's old velocity $\overrightarrow{v}(t)$, its own *cognitive learning* part, ϕ_1 ($\overrightarrow{p}(t)$ - $\overrightarrow{x}(t)$) and *social learning* part, ϕ_2 ($\overrightarrow{g}(t)$ - $\overrightarrow{x}(t)$).

After having calculated the velocities and position for the next time step t+1, the first iteration of the algorithm is completed. Typically, this process is iterated for a certain number of time steps, or until some acceptable solution has been found by the algorithm. Here the pseudo-code for the PSO algorithm is presented. During the search, the particles exchange information about their positions and fitness values. This communication results in that the swarm learns and refine its knowledge about the search, and moves towards the good search space areas. This is analogous to flocks of birds flying and searching for food, to social insects such as bees and ants when foraging or nesting, and to humans who affect the minds of each other by interacting socially.

Program of Particle Swarm Optimization Algorithm:

Set t = 0;

Initialize ϕ_1, ϕ_2, V_{max} *and define N;*

$\forall p \in S$ *: Initialize* $\overrightarrow{x}(t)$, $\overrightarrow{v}(t)$, $\overrightarrow{p}(t)$, *and* $\overrightarrow{g}(t)$ *as described.*

While

{

Min. is not reached or Iterations not exhausted

$\forall p \in S$: *Calculate* $\overrightarrow{v}(t+1)$ *and* $\overrightarrow{x}(t+1)$ *using 1.8 and 1.9.*

$\forall p \in S$: *Update* $\overrightarrow{p}(t+1)$ *with* $\overrightarrow{x}(t+1)$ *if f (*$\overrightarrow{x}(t+1)$*) is better than f(*$\overrightarrow{x}(t)$*)*

$\forall p \in S$: *Update* $\overrightarrow{g}(t+1)$ *with* $\overrightarrow{p}(t+1)$ *in N(p).*

}

As an emergent result of the two equations 1.8 and 1.9, the swarm as a whole will identify and approach the good areas of the search space in a self-organized structure based on comparison to and imitation of the particles to each other.

On the algorithmic level, the main strength of the PSO is its fast convergence, which compares favorably to many EAs [108]. However, it has three major weaknesses:

1. It cannot dynamically adjust its velocities when fine-tuning a found optimum, and hence the convergence rate decreases dramatically in the close vicinity of optima [8].

2. On hard problems with many optima, its fast convergence rate may result in premature convergence [78].

3. The number of PSO parameters to tune is critically large [78].

However, there are a number of techniques available in the literature that address these problems and try to minimize them. For instance, to control the convergence, the inertia-weight model by Eberhart and Shi can be used [135]. We too have used this technique in this thesis. This technique multiplies the velocity of the current time-step t with a factor called the inertia weight, w, in the calculation of the new

velocity at $t+1$, i.e.:

$$w * \overrightarrow{v}(t+1) \quad = \quad \overrightarrow{v}(t) + \phi_1(\overrightarrow{p}(t) - \overrightarrow{x}(t)) + \phi_2(\overrightarrow{g}(t) - \overrightarrow{x}(t)) \qquad (1.10)$$

Where $w \in [0,1]$ is to enforce convergence. The reducing factor w is only multiplied with $\overrightarrow{v}(t)$.

Another technique is using the Maurice-Clerc model involving the constriction factor, K [35]. The resulting equation for the velocity is given as:

$$\overrightarrow{v}(t+1) \quad = \quad K * (\overrightarrow{v}(t) + \phi_1(\overrightarrow{p}(t) - \overrightarrow{x}(t)) + \phi_2(\overrightarrow{g}(t) - \overrightarrow{x}(t))) \qquad (1.11)$$

Other weaknesses of PSO are not critical for the applications in this thesis but they can be avoided by using well-known techniques. Premature convergence can be avoided by using hybrid-PSO algorithms, i.e. in combination with GAs, as described by [91]. The convergence can be speeded up by using techniques to improve the particles' trajectories as given in [80].

1.4 Statement of Problem

The aim of this thesis is to develop a new algorithm that can be used for effective control of processes and systems. This advanced control algorithm is realized by utilizing the Model Predictive Control technique described in the Section 1.2. This technique will require the optimization of a quadratic criterion or objective function. This optimization is done by using the heuristic technique of Particle Swarm Optimization

described in Section 1.3. PSO technique has several advantages over other heuristics that have been already discussed. Particularly, the PSO is used to provide quick results while optimizing complex objective functions with multiple objectives and any process constraints.

The algorithm can be represented by the following block diagram:

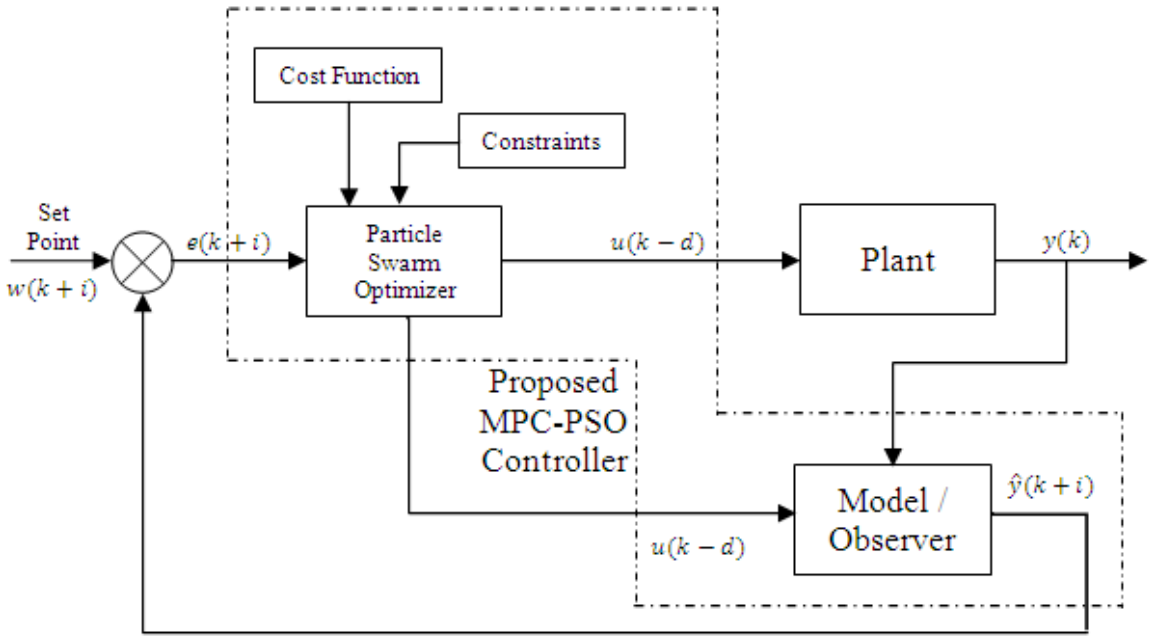

Figure 1.4: Structure of Proposed MPC-PSO Controller

The selection of a feasible and realizable objective function is also very important. Generally, the objective functions include some penalty terms, like the weighted sums of control signals (inputs) over the control horizon and weighted sums of the rate of change of control signals over the prediction horizon, in addition to the weighted sum of errors over the prediction horizon. Also, constraints are taken into account which are due to the physical limitations on the actuators. So, a generalized objective

function that is used is given as follows:

$$J = \sum_{i=1}^{H_p} e(k+i)^T Q e(k+i) + \sum_{i=1}^{H_c} \Delta u(k+i)^T R \Delta u(k+i) + \sum_{i=1}^{H_p} u(k+i)^T S u(k+i) \quad (1.12)$$

Subjected to the following constraints:

$$u_{min} \leq u(k+i) \leq u_{max} \quad (1.13)$$

$$\Delta u_{min} \leq \Delta u(k+i) \leq \Delta u_{max} \quad (1.14)$$

$$y_{min} \leq y(k+i) \leq y_{max} \quad (1.15)$$

In Equation 2.20 Q, R and S are the weights on the prediction error, *e(k)*, change in the input, Δu, and magnitude of the input, u, respectively. The prediction error is defined as,

$$e = w(k) - \hat{y}(k) \quad (1.16)$$

Where $w(k)$ is the reference or the desired set point.

The process models used in this thesis are the Linear and Non-Linear State Space Models. Some linear and nonlinear examples are taken at the start for studying, but the main concentration is the optimal control of linear and nonlinear, SISO and MIMO, multi-order, power systems.

1.5 Thesis Organization and Scope

This thesis is organized as follows: In the first chapter, the importance of MPC is emphasized and the motivation for this work is given. The algorithm is also described briefly. The Particle Swarm Optimization Technique is also discussed. Concise but comprehensive literature reviews for both MPC and PSO are also given. It is seen that heuristic based MPC algorithms are rarely used in the literature and they have not been previously applied to any power systems.

In the second chapter, the MPC technique is explored in detail along with the necessary concepts associated with it, like description of the process models, process constraints, advantages and disadvantages etc. The MPC-PSO algorithm is also proposed in this section and explained. The proposed controller is applied to a nonlinear control valve to illustrate the use and explain it through an example.

The proposed controller is then applied to three major nonlinear power systems. These systems are high order, MIMO and nonlinear. Chapters 3 - 5 detail these applications. Three power systems selected for applying the proposed controller. The details are given in Figure 1.5.

In Chapter 6, the results are summarized and special topics are discussed. The foreseen future improvements that can be done to the proposed controller and its applications are also discussed.

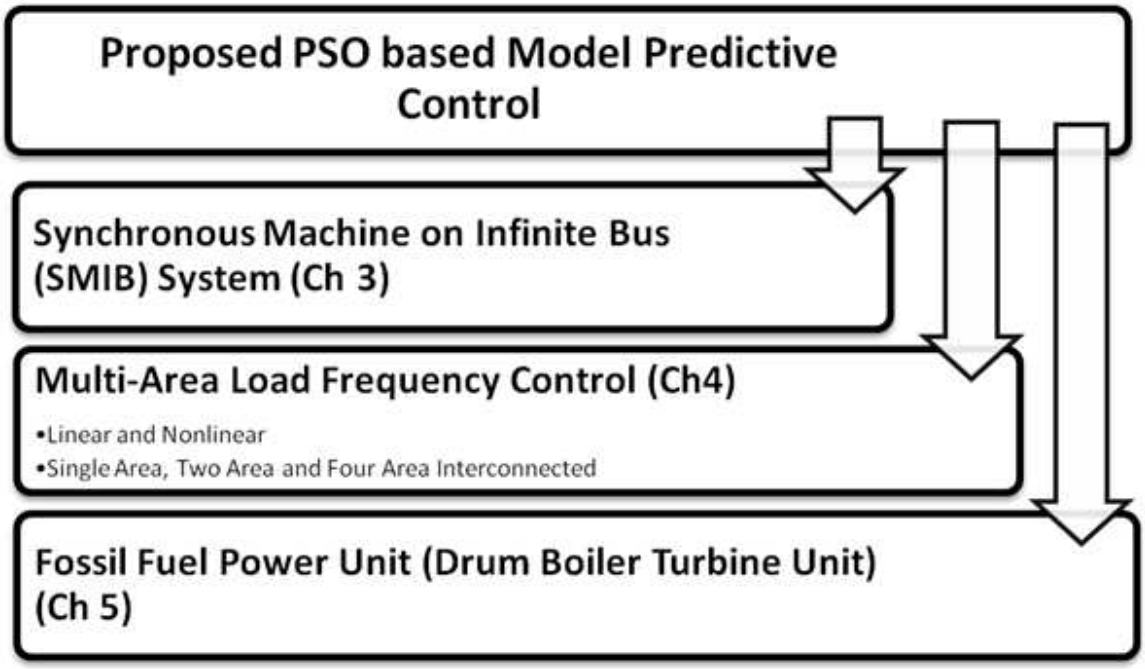

Figure 1.5: Power System Dynamics Tested with the Proposed Controller

CHAPTER 2

PSO BASED MODEL PREDICTIVE CONTROL: MPC-PSO

In Chapter 1, Model Predictive Control was introduced and some detail was given, including an introduction to the algorithm. In this chapter, the MPC concept is described in more detail. The proposed MPC-PSO controller is also introduced and explained. Then it is applied for the nonlinear control of an automatic valve to illustrate its use and understand it better.

2.1 The Predictive Controller Concept

Model Predictive Control refers to a class of algorithms that compute a sequence of signals or manipulated variable adjustments, in order to optimize the future behavior

of a plant. The optimal sequence is generated by utilizing a model of the process. The model of the system is any entity that describes the input and output relations and any type of model can be used. These models are given in Section 2.2.1. Naturally, these models can be linear or non-linear. Also, if the process is subjected to disturbances, noise or variations, these can be incorporated to the process models in the form of disturbance or noise models. This will allow the effect of disturbances on the predicted process to be taken into account. MPC also incorporates process constraints in the prediction, as mentioned earlier. All of these qualities constitute the Model Predictive Controller Concept [74].

To deeply understand the process, consider Figure 1.2 again. If the current time is denoted by k, $u(k)$, $y(k)$ and $\hat{y}(k)$ denote the controller output, the process output and the predicted process output respectively at the time k. w is the desired process output or the set point. Now, we define,

$$u = [u(k), u(k+1), ..., u(k+H_p-1)]^T \tag{2.1}$$

$$\hat{y} = [\hat{y}(k+1), \hat{y}(k+2), ..., \hat{y}(k+H_p)]^T \tag{2.2}$$

$$w = [w(k+1), w(k+2), ..., w(k+H_p)]^T \tag{2.3}$$

Here, H_p is the prediction horizon i.e., the time in the future up to which the output is predicted using the model. Using this predictive output, the predictive controller computes the future controller output sequence u as shown in Figure 1.2 such that the predicted output of the process, $\hat{y}$ is as close to the desired process output, w, as possible. This desired process is called the *reference* or the *reference trajectory*.

When the controller output sequence, *u(k)* is obtained in the above way for controlling the process in the next H_p samples, only the first element of *u(k)* is used to control the process instead of the complete controller output sequence. At the next sample, k+1, this whole process is repeated using the latest measured information. This is called the *receding horizon* principle [74]. Assuming that there are no disturbances or modeling errors, the predicted process output, $\hat{y}(k+1)$ is exactly equal to the actual process output. Again, a future controller output sequence, *u(k)*, is calculated such that the predicted output is close to the reference trajectory. Generally, this controller output sequence is different from the one obtained at the previous sample. The reason of using the receding horizon technique is that it allows us to compensate for future disturbance or modeling errors. For example, if for a case at k, due to a disturbance or modeling mismatch, the predicted process output, $\hat{y}(k+1)$ is not equal to the process output, *y(k)*, then it makes sense to start predictions from the measure process output at time k+1, rather than the process output predicted at the previous sample, k. The predicted process output is now corrected for disturbances and modeling errors activating a feedback mechanism. Resulting from the receding horizon approach, the horizon H_p shifts one sample into the future at every sample instant, predicting the process output again.

2.1.1 Summary of Model Predictive Control Algorithm

The Model Predictive Control algorithm can be summarily described to generally have the following three steps [114].

1. Explicit use of a model to predict the process output along a future time horizon (Prediction Horizon, H_p)

2. Calculation of a control sequence along a future time horizon (Control Horizon, H_c), to optimize a performance index.

3. A receding horizon strategy, so that at each instant the horizon is moved towards the future which involves the application of the first control signal of the sequence calculated at each step which is illustrated in Figure 1.2.

2.2 Elements of the Predictive Controller Design

In this section, the components that build up the predictive controller are discussed in more detail.

2.2.1 Process Models

The model of the process is the heart of the Model Predictive Controller concept. All MPCs explicitly use a model of the plant to be controlled to determine the future behavior or outputs of that plant or process. For an ideal MPC system, the model should match the process exactly. Practically, this is rarely that case and generally, identification techniques are used to obtain a workable model of the process to be controlled. Therefore, generally, all process models are derived from plant testing or system identification.

Process models are linear as well as non-linear. Since most of the industrial and

power processes are nonlinear, they have been emphasized and worked upon in this thesis. However, a list of the process models found in literature is also given because they can be used with MPC as well.

Linear State Space Model

Every linear lumped system can be described by a set of equations of the form [40]:

$$\dot{x}(t) = \mathbf{A}x(t) + \mathbf{B}u(t) \tag{2.4}$$

$$y(t) = \mathbf{C}x(t) + \mathbf{D}u(t) \tag{2.5}$$

For a system with the following parameters:

- Inputs = p
- Outputs = q
- State Variables = n

The size of the constant matrices, A, B, C, and D are:

- $\mathbf{A}$ = n x n
- $\mathbf{B}$ = n x p
- $\mathbf{C}$ = q x n
- $\mathbf{D}$ = q x p

While in the equations 2.4 and 2.5, x are the states, $\dot{x}$ represents the derivative of the states, u is the input and y is the output of the process.

This concept can be extended to give the i-step ahead prediction of the output of the process in a simple manner:

$$\dot{x}(k+1|k) = \mathbf{A}x(k|k) + \mathbf{B}u(k|k) \tag{2.6}$$

$$y(k+1|k) = \mathbf{C}x(k|k) + \mathbf{D}u(k|k) \tag{2.7}$$

Where *x(k + 1 | k)* means the prediction of x at time *k + 1* given the information at time k.

Nonlinear State Space Model

Most physical systems, including the ones studied in this thesis are non-linear. Systems can also be time varying. Some of these systems can be described in state space form by nonlinear differential equations which are of the form:

$$\dot{x}(k+1|k) = \mathbf{h}(x(k|k), u(k|k), k) \tag{2.8}$$

$$y(k+1|k) = \mathbf{f}(x(k|k), u(k|k), k) \tag{2.9}$$

where **h** and **f** are nonlinear functions. The behavior of such equations can be very complicated.

Other Models

Other dynamic models of the systems that can be used with MPC are are follows:

1. Linear Models:
    - Impulse Response Model
    - Step Response Model
    - Transfer Function Model
2. Nonlinear Models:
    - Hammerstein Model
    - Wiener Model
    - NARMAX (Nonlinear Auto Regressive and Moving Average with Exogenous Inputs)
3. Disturbance Models

2.2.2 Cost Functions

To determine the health of the tracking (predicted process output, $\hat{y}(k)$ tracking the reference trajectory, *w(k)*), a criterion function or cost function is used. Typically, such a function is a function of $\hat{y}$, w and u. A simple criterion function is given in Equation 2.10 and it is,

$$J = \sum_{i=1}^{H_p} [\hat{y}(k+i) - w(k+i)]^2 \tag{2.10}$$

In this criterion, there is no involvement of u. Other criterion functions can be obtained by augmenting different penalty terms to this criterion function. These penalties usually involve the input, u and the rate of change of the input, Δu. These quantities are penalized by weighting matrices when they exceed a certain desired threshold. A more comprehensive criterion function is this,

$$J = \sum_{i=1}^{H_p} e(k+i)^T Q e(k+i) + \sum_{i=1}^{H_c} \Delta u(k+i)^T R \Delta u(k+i) + \sum_{i=1}^{H_p} u(k+i)^T S u(k+i) \quad (2.11)$$

where Q, R and S are the weighting matrices, H_c is the control horizon and e is the error between the desired output and the predicted output. i.e.,

$$e = w(k) - \hat{y}(k) \quad (2.12)$$

Now the controller output sequence u_{opt} over the prediction horizon is obtained by the minimization of J with respect to u. i.e.,

$$u_{opt} = min_u J \quad (2.13)$$

Then, u_{opt} is optimal with respect to the criterion function that is minimized. As a result, the future tracking error is minimized. If the model is exactly identical to the process and there are no disturbance or constraints, then the process will track the reference trajectory exactly on each of the sampling instants.

2.2.3 Process Constraints

Most practical control problems are dominated by process constraints and nonlinearities. These constraints are discussed here before moving on to the discussion of the proposed MPC-PSO controller, to enable easier understanding of the literature ahead.

For constrained model predictive control of a physical system, some criteria must be satisfied along with the minimization of the quadratic cost function. These conditions/criteria are known as *constraints*. The most common constraints are constraints on the manipulated and/or state variables. These constraints can make even a linear system nonlinear. These nonlinearities can be "weak" or "strong" and it may be that a linear controller design might be working well for a "weakly nonlinear" system, it will most probably for a "strongly nonlinear" system [6].

With regards to process constraints, they are present in the manipulated variables in almost all processes because of the physical limitations of the actuators which cannot exceeded. For safe plant operation, states such as velocity, acceleration, temperature, pressure, revolutions, etc must also be constrained. Constraints can also be used to represent the performance objectives of the controllers. Although most control constraints should be respected throughout the operation i.e. *hard constraints*, some times, especially during the case when the system is subjected to unexpected disturbances, it may be unavoidable to exceed some state constraints i.e. *soft constraints* [113]. Hard constraints are usually imposed on the input to the process while soft constraints are usually implemented on the output of the process. Obviously, it is intended to avoid violations of the soft constraints as well to ensure optimal plant operation,

however, temporarily violations can be allowed for the satisfaction of other criteria, and the magnitude of the violation is generally subjected to a quadratic penalty in the objective function. In MPC, the use of hard constraints is generally avoided because a disturbance can easily cause such a controller to become non-feasible [113]. Various types of constraints are given below. More or less, these constraints are similar, but it is worth mentioning them here because they are frequently categorized in the literature as well [40].

- Equality Constraints
- Non-Equality Constraints
- End Constraints
- Level Constraints
- Rate Constraints

Equality Constraints

Equality constraints refer to the equality of some input or output to a specified value. For example,

$$k(x(t), u(t), t) = 0 \tag{2.14}$$

An application of this constraint is in the regulator problem where the output of the process must be maintained at a fix value.

Inequality Constraints

Inequality constraints refer to the conditions that some input or output must be greater or lesser than some specified value. For example,

$$k(x(t), u(t), t) \geq 0 \tag{2.15}$$

$$k(x(t), u(t), t) \leq 0 \tag{2.16}$$

This type of constraint also finds applications in many problems.

Level Constraints

Level constraints refer to the aggregation of equality and inequality constraints i.e. the condition when the controller is restricted between two values, $[u_{max}\ u_{min}]$. For example,

$$u_{min}(k) \leq u(k) \leq u_{max}(k) \tag{2.17}$$

Rate Constraints

Rate constraints refer to the condition when the change of the controller output per sample is limited between two values, $[\Delta u_{max}\ \Delta u_{min}]$. For example,

$$\Delta u_{min}(k) \leq \Delta u(k) \leq \Delta u_{max}(k) \tag{2.18}$$

This type of constraints are usually applied to avoid large changes in the input moves to limit large changes in the output of the process. Rate constraint is used extensively in this thesis to implement constraints on change in control effort, Δu.

Constrained MPC problems are found in literature in the publications of Palazoglu [103] and [102], Cheng [32], and Rossiter [124] and many others. Chapter 8 of Rossiter's book [73] gives a detailed account on constraints and their handling while using MPC.

2.3 PSO based Model Predictive Control: MPC-PSO

In this thesis a new PSO-based model predictive controller is proposed. The block diagram of the proposed controller is shown in Figure 2.1. The purpose of the controller is to use the process model to search for the best control signals to be applied. However, this must be done while satisfying some constraints and optimizing some cost function. The process model can be of any type and in this thesis it is generally a nonlinear state space representation of the system to be controlled. The algorithm is described in Figure 2.2 in the form of a flowchart and it is explained in the next section.

2.3.1 Implementation of MPC-PSO

The algorithm proposed in Section 1.4 is implemented in MATLAB. It consists of a new MPC technique to control processes by incorporating the Particle Swarm Optimization algorithm. The particles are initialized at the started by assigning them random values. They can also be initialized randomly around a previously obtained optimal

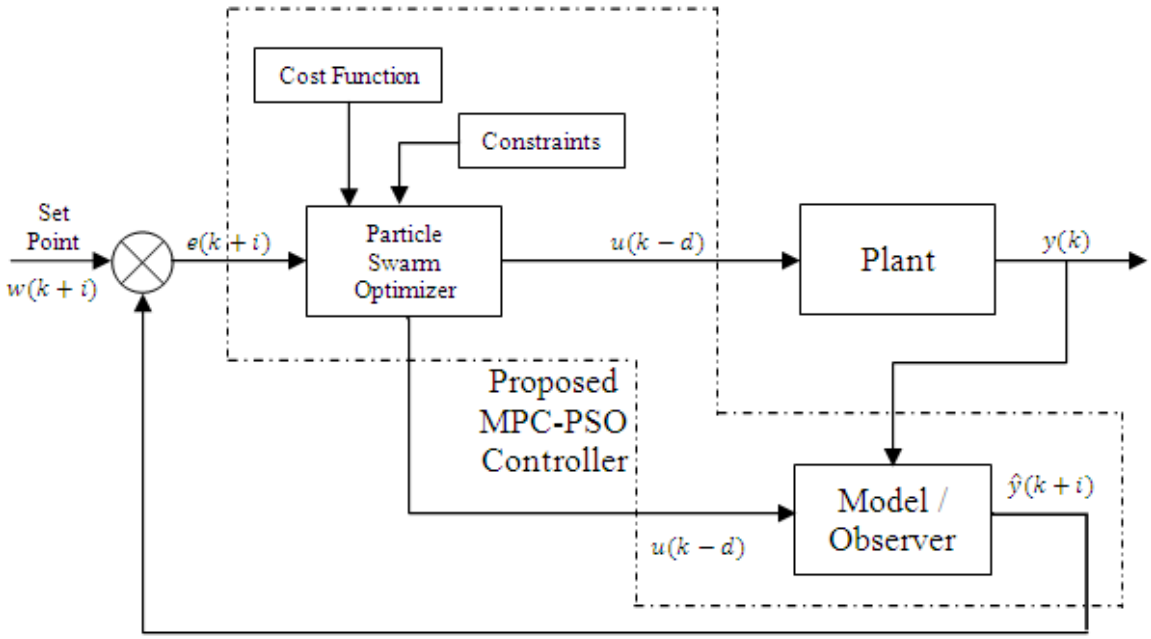

Figure 2.1: Structure of Proposed MPC-PSO Controller

control signal within a certain range, which will be typically defined by the rate constraint on that control signal. A set of inputs for the process is generated and is applied to the model. The output of the model is obtained and a cost function is evaluated based on these outputs. The inverse of the cost determines the fitness function. The fitness function used in this algorithm is the following:

$$fitness = \frac{1}{|J|} \tag{2.19}$$

Where J is the cost function or the performance index. This will be described in detail later.

The PSO algorithm then finds the optimal input sequence which physically consists of the control moves. The particle values are updated with these values and applied

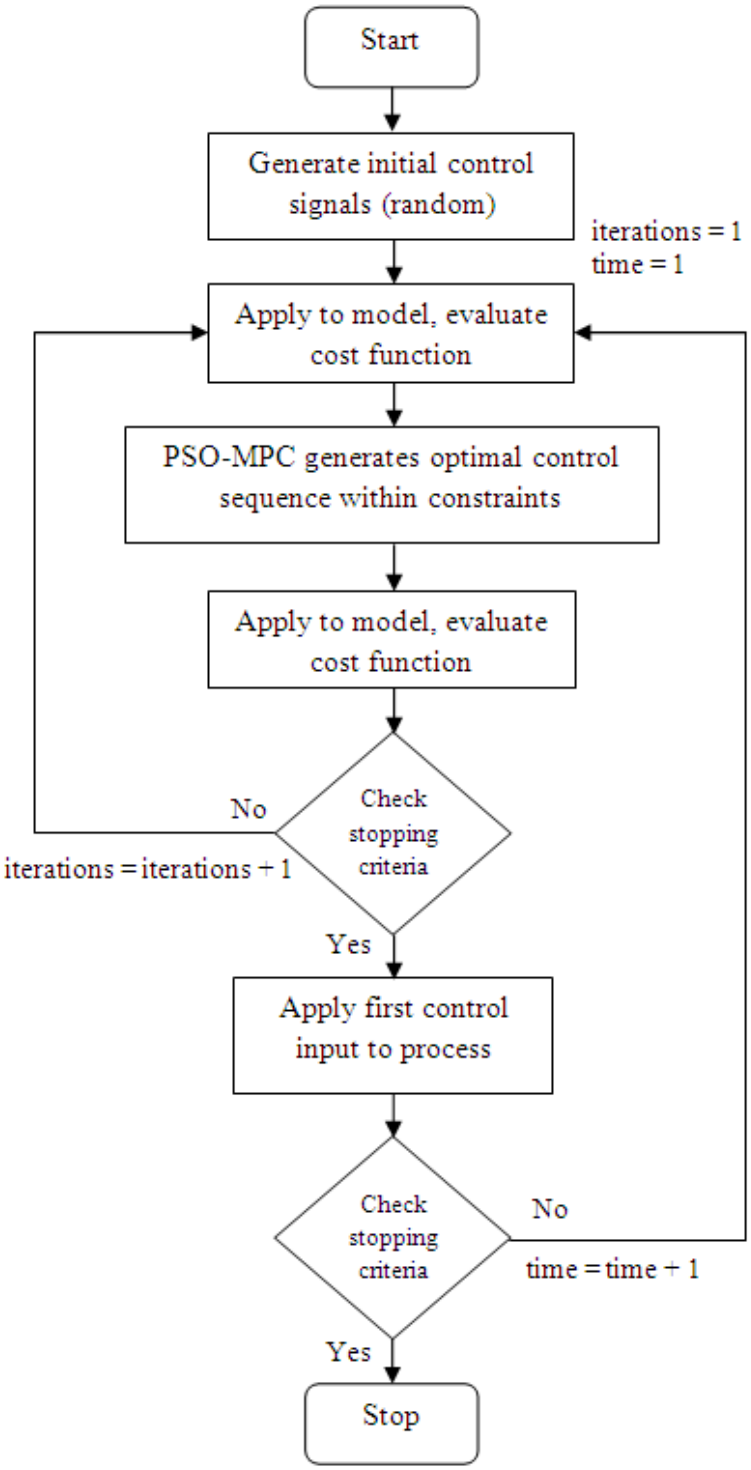

Figure 2.2: Flowchart of Proposed PSO-based MPC

to the model again. This is done a certain number of times.

The PSO algorithm uses the real direct values obtained from the process and does not require conversion into another number system or integers. This gives obvious advantages of practicality, less computation time and straightforward understanding and interpretation.

The number of particles represent the prediction horizon, H_p. If there are more than one inputs, i.e. we have a MIMO system for example, the number of particles is increased proportionally. So for a two input system, the number of particles is doubled. Note that the particles are not to be confused with the population of the particle swarm. For each particle there is a certain population in the swarm. So, if the prediction horizon is 3 for example, and the population is 20, the swarm size will be 3x20. So the total population of the particle swarm will be 60. Obviously, the first 20 of them will be for the present instant, while the next 40 will be for the next two instants or samples and so on.

After the cost function is evaluated, an optimal control signal is selected. This optimal control signal is the one which gives the least cost, or the most fitness. This is applied to the plant itself. Thus the plant moves ahead by one sample.

This whole process from the random initialization of the swarm to this last step is repeated again for the plant, which has by now advanced one step due to the application of the optimal control signal. This can be clearly seen in the flowchart in Figure 2.2. After the system been run for a certain time, the simulation stops.

Now the performance index or the cost function is discussed. This is given earlier

in Equation 2.20 and is evaluated as the weighted sum of square of errors between actually and predicted outputs over a finite prediction horizon, H_p. Incorporated into the performance index is the weighted sum of the square of the change in inputs over the control horizon, H_c, and the weighted sum of the square of the input moves over the prediction horizon. The equation for the performance index is thus,

$$J = \sum_{i=1}^{H_p} e(k+i)^T Q e(k+i) + \sum_{i=1}^{H_c} \Delta u(k+i)^T R \Delta u(k+i) + \sum_{i=1}^{H_p} u(k+i)^T S u(k+i) \quad (2.20)$$

Subjected to the following constraints:

$$u_{min} \leq u(k+i) \leq u_{max} \quad (2.21)$$

$$\Delta u_{min} \leq \Delta u(k+i) \leq \Delta u_{max} \quad (2.22)$$

$$y_{min} \leq y(k+i) \leq y_{max} \quad (2.23)$$

In Equation 2.20 Q, R and S are the weights on the prediction error, *e(k)*, change in the input, Δu, and magnitude of the input, u, respectively. The prediction error is defined as,

$$e = w(k) - \hat{y}(k) \quad (2.24)$$

Where $w(k)$ is the reference or the desired set point.

2.3.2 Constraint Formulation

Constraints and their handling ability are very important to the MPC-PSO algorithm. Constraints are applied to both the process inputs, i.e. the control signals and also to the rate of change of these inputs.

The constraints on the inputs are applied at the time of generating the population of the particle swarm, which is usually random. The swarms are initialized within the ranges allowable keeping in view the constraints. Another way to implement constraint on the input is by capping the value of input to the maximum or minimum allowable value when a violation is detected. This is done by putting a check on the generated control signal during each iteration. The psuedocode for this technique can be written as:

$$IF\{current\_input\} > \{maximum\_allowable\_control\};\ \{current\_input\} = \{maximum\_allowable\_control\}$$

$$ELSEIF\{current\_input\} < \{minimum\_allowable\_control\};\ \{current\_input\} = \{minimum\_allowable\_control\}$$

$$ELSE;\ \{current\_input\} = \{current\_input\}$$

The constraints on the rate of change of inputs are applied by comparing the input at each time instant, *k*, with the input at the previous time instant, *k-1*. If the difference of the two inputs, Δu is violating the constraints, the input is set to the extreme value allowable by the constraint. Therefore, if Δu is higher than the desired range, u at the instant k is set to the maximum value and vice versa. This prevents

the rate of change of inputs from violating the constraints. This can be coded as:

$$IF\{current\_input - previous\_applied\_input\} \geq \Delta u_{max};\ \{current\_input\} = \{previous\_applied\_input\} + \Delta u_{max}$$

$$ELSEIF\{current\_input - previous\_applied\_input\} \leq \Delta u_{min};\ \{current\_input\} = \{previous\_applied\_input\} - |\Delta u_{min}|$$

$$ELSE;\ \{\text{current\_input}\} = \{\text{current\_input}\}$$

Some processes have constraints on particular states. This is also taking care of in a similar way by checking the particular states at each time instant k for violations and if there is a violation, the cost function is set to a very high value thus eliminating that state and input sequence to be applied to the system. This can be applied as:

$$IF\{observed\_state\} > \{maximum\_allowable\_value\};\ cost = 10^{10};$$

$$ELSEIF\{observed\_state\} < \{minimum\_allowable\_value\};\ cost = 10^{10};$$

2.4 MPC-PSO Explained through Nonlinear Control of a Valve

To understand the concept of MPC-PSO better, a simple example of a nonlinear control valve is presented.

Control valves are one of the basic components of process and power systems control. The control valve is essentially an opening with adjustable volume and is

widely used to control the flow of fluids. It consists of an actuator, a valve body and a valve plug. The actuator translates the control signal into motion of a piston which moves the valve plug resulting in a specific volume through which the liquid is allowed to flow. This controls the flow rate of the liquid through the valve.

The dynamics of a typical control valve are described as a Wiener Model in [157]. Wiener Model is a specialized nonlinear model that consists of a linear dynamic block followed by a nonlinear zero memory block and is explained in detail by Palazoglu in [103]. Details of Wiener model or modeling of a control valve are not considered a part of this discussion.

The model of the control valve is given below and is described by the linear dynamics in Equation 2.25 and Wiener nonlinearity in Equation 2.26.

$$x(k) = \frac{0.0616q^{-1} + 0.0543q^{-2}}{1 - 1.5714q^{-1} + 0.6873q^{-2}}u(k) \tag{2.25}$$

$$y(k) = \frac{x(k)}{\sqrt{0.10 + 0.90x^2(k)}} \tag{2.26}$$

where *u(k)* is the control signal (pressure), *x(k)* is the stem position, and *y(k)* is the flow through the valve. *y(k)* is the control variable.

The nonlinear characteristics of the control valve described by the Equation 2.26 are given in Figure 2.3. It is seen that the behavior is most nonlinear in the range of -0.4 $\leq x(k) \leq$ 0.4.

The proposed controller is applied to the nonlinear model of the control valve and is tested for both regulatory as well as servo behavior. The PSO parameters are taken as:

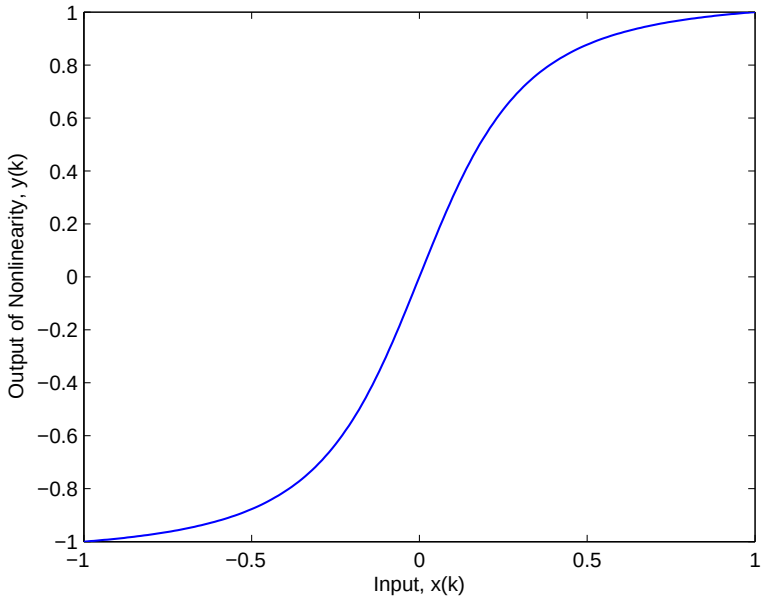

Figure 2.3: Nonlinear Characteristic of a Control Valve

- Population = 20
- Iteration = 500
- $w_{max} = 0.9$
- $w_{min} = 0.4$
- $c_1 = 2.04$
- $c_2 = 2.04$

The prediction horizon is taken to be 3.

The cost function is given comprehensively in the Equation 2.20 as follows:

$$J = \sum_{i=1}^{H_p} e(k+i)^T Q e(k+i) + \sum_{i=1}^{H_c} \Delta u(k+i)^T R \Delta u(k+i) + \sum_{i=1}^{H_p} u(k+i)^T S u(k+i)$$

For this example, $Q = t$, $R = 0$, and S = 0.01. $Q = t$ means that the weight on the error increases with time. So gradually the emphasis on minimizing the error increases significantly than the emphasis on minimizing the control. The constraint on Δ is applied continuously but is not built in to the cost function. Instead, it is implemented as a hard constraint on the system due to the physical dynamics of the valve actuator, and can not be violated. This is formulated as described in Section 2.3.2.

The control effort, u in this example is thus taken to be constrained in two ways:

- $0 \leq \text{u} \leq 1$
- $\text{-}0.5 \leq \Delta\text{u} \leq 0.5$

This means that the control signal is constrained between a minimum and maximum value of 0 and 1 respectively and cannot vary by more than 0.5 between samples in any case.

All simulations in this thesis are done on MATLAB on an Intel 2Ghz Core2Duo machine with 2GB of RAM. The programming is directly done in MATLAB without using any toolboxes.

The system is initialized at 0. The sampling time is taken as 0.1s. The swarm is initialized as a 3x20 matrix with random values between 0 and 1. The optimal control effort is also initialized at zero, and Δu is set at 0.5. The model of the system is explicitly programmed as well. The system is simulated for the first time with these values and the first optimal control signal is search for and selected using the cost function as described in Section 2.3.1. The PSO algorithm is run for 500 iterations

which is enough to ensure that optimal control effort is the best one for this instant. At the same time the optimal control efforts for the prediction horizon (next two samples) is also obtained. This will be the scope of work for the controller at each sample.

The first control signal is applied to the control valve and the system is advanced to the next state. This state is now the initial condition for the next sample. This process goes on and optimal control signal is continuously generated and applied to the system. At the end of a certain time (30s) the simulation is stopped. The next two sections detail the results of these simulations for regulatory and servo cases.

2.4.1 Regulatory Control of Control Valve Using MPC-PSO

First the proposed controller is applied to the nonlinear control valve for regulation. The objective is to track a given set point. The set point is switched from 0 to 0.5 as seen in Figure 2.4. The proposed controller applies the optimal control effort to bring the valve position to the required set point, i.e. 50% open. The valve position is also given in Figure 2.4. It is seen that there is an overshoot of 0.2 and the valve position momentarily goes up to 70% open. Then it quickly decreases to reach the required set point. So at about 12s, the valve is at the required set point, i.e. 2 seconds after the command is sent. The applied optimal control effort is seen in Figure 2.5. It is seen that the control effort is within the minimum and maximum limits of $[0, 1]$ and the rate of change is also a maximum of 0.5 between samples.

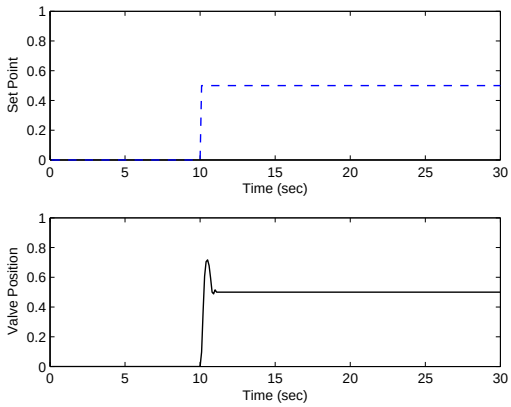

Figure 2.4: Set Point Tracking of Control Valve for Proposed Controller

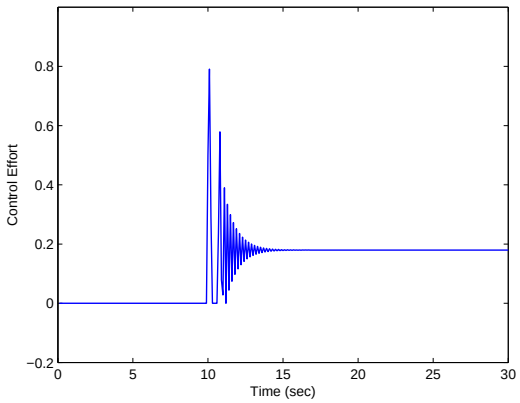

Figure 2.5: Control Effort for Set Point Tracking of Control Valve

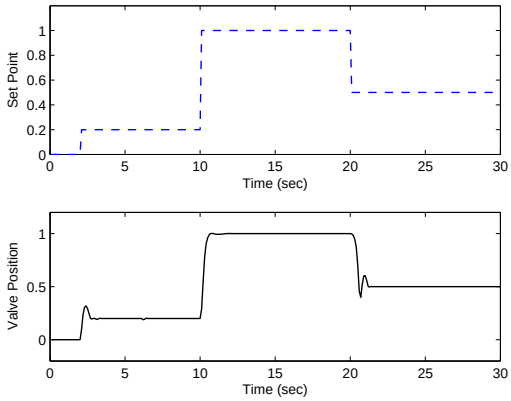

Figure 2.6: Servo Behavior of Control Valve for Proposed Controller

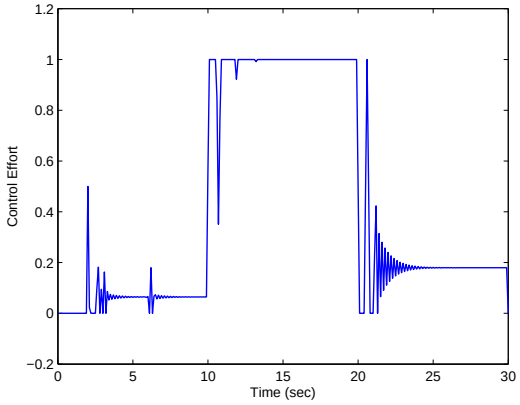

Figure 2.7: Control Effort for Servo Behavior of Control Valve

2.4.2 Servo Control of Control Valve Using MPC-PSO

The proposed control is also applied for the servo control of the nonlinear valve. The set point is varied as shown in Figure 2.6. The valve position follows the given set points and the behavior is seen in the same figure. The figure is self explanatory. The control effort for this case is shown in Figure 2.7.

2.5 Advantages and Disadvantages of MPC

The advantages and disadvantages of MPC are discussed in detail in Soeterboek's book [115]. Some of the advantages are:

- The concept of predictive control is not restricted to single-input, single-output processes. Predictive controllers can be derived for and applied to multi-input, multi-output processes.

- Predictive controllers can be applied to linear and nonlinear processes.

- Predictive control is the only methodology that can handle process constraints in a systematic way during the design of the controller.

- The concept of predictive control can be used to control a wide variety of processes without the designer having to take special precautions.

- In a natural way, feed-forward action can be introduced for compensation of measurable disturbances and for tracking reference trajectories.

- Because predictive controllers make use of predictions, pre-scheduled reference trajectories (for example, used in robot control) or set points can be dealt with.

- Predictive control is an open methodology. That is, within the framework of predictive control there are many ways to design a predictive controller. As a result, different predictive controllers, each with different properties, have been proposed in the literature over the last 30 years. Many of them are mentioned in the literature review in Chapter 1.

Some disadvantages found in the MPC technology are [115]:

- Since predictive controllers belong to the class of model-based controller design methods, a model of the process must be available. In general, while designing a control system two phases can be distinguished: modeling and controller design. Predictive control provides only the solution for the controller design part. A model of the process must be obtained by other methods.

- A second drawback is due to the fact that the predictive control concept is an open methodology. It has already been mentioned that due to this, different predictive controllers can be derived having different properties. Although the differences between these controllers seem rather small, these small differences can yield very different behavior of the closed-loop systems. As a result, it can be quite difficult to select which predictive controller must or can be used to solve a particular control problem. One cannot afford the risk and expense of designing a control system that might not work with another process and the cost of which cannot therefore be spread over a large number of applications.

2.6 Conclusion

The predictive controller concept is given in detail. It is seen that the MPC can incorporate a variety of cost functions, and constraints, as well as use any explicit model of the system available. The proposed controller is also now well understood. The application of the proposed controller is illustrated on the example of a nonlinear control valve and the results show the the proposed controller can successfully control

the nonlinear system directly, without the need of any approximation or linearization. Furthermore, the optimal control effort is generated while obeying all the constraints and limitations of the system as well as physical limitation on the actuator. The proposed controller works well for both regulatory as well as servo applications.

CHAPTER 3

APPLICATION OF MPC-PSO TO SYNCHRONOUS MACHINE ON INFINITE BUS (SMIB) SYSTEM

In this chapter, the Synchronous Machine connected to an Infinite Bus (SMIB) system is studied and the proposed MPC-PSO controller is applied directly to its nonlinear model. Various scenarios of operation are considered.

3.1 Introduction

In the literature, many methods are used for the control of the SMIB system. Various approaches to design the controller are available, including Classical Control [39], Optimal Control [171], Adaptive Control [101], Variable Structure Control (VSC) [24], and Intelligent Control [66]. Traditionally, all of these methods involve linearization,

at least to some degree if not complete [170], [138] or some sort of complex nonlinear transformations to reduce the order of the system [139], [143], and [69]. Therefore, some adaptive control methods were also suggested by Pierre [111] and Ghosh [54].

Al-Musabi [4] and his advisors, Al-Hamouz and Al-Duwaish proposed a newly design VSC for SMIB system utilizing iterative heuristic optimization techniques (GA, PSO and TS) to provide a simpler, more systematic method with no need for complex approximations. This enabled direct application to the VSC design to the nonlinear model without undergoing bothersome transformations. It was successfully applied to the model given by Matthews in [95] and showed significant improvements compared to previous work on this subject.

As in preceding literature, the primary objective is to drive the SMIB system from an already perturbed, possibly unstable state to desired setpoints. Furthermore, some more control objectives involving real errors in the system are explored and the control of the system in the event of real component faults is also tried. This contribution is very important to stamp the power of MPC-PSO since it will show that the algorithm can bring complex nonlinear unstable systems with real faults to a desired set point without the need of approximations, linearizations, transformations or model reduction. Another feature of this contribution will be that the system will be stabilized very quickly.

First we will study the nonlinear model of the SMIB system in the next section followed by its control in the proceeding sections.

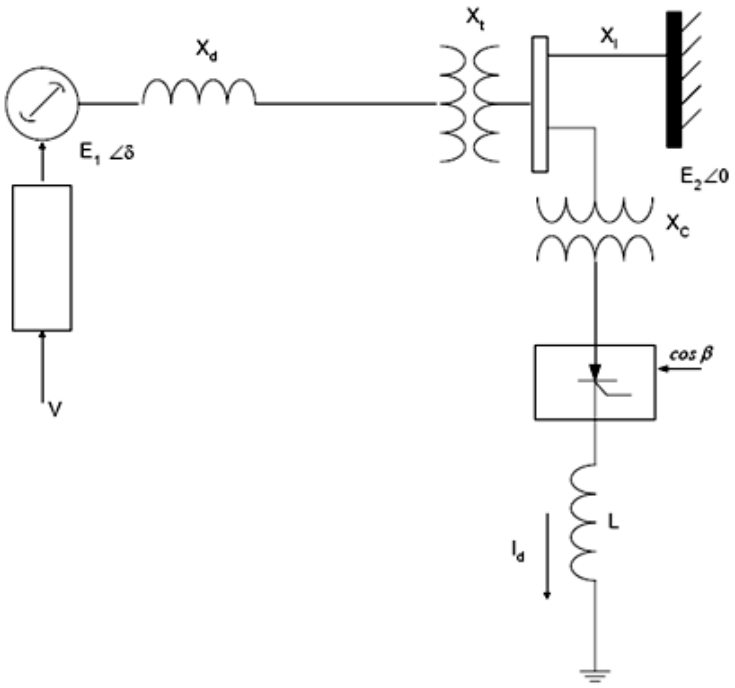

Figure 3.1: Diagram of Synchronous Machine Infinite Bus (SMIB) Power System

3.2 Nonlinear Model of the SMIB System

In [95], a nonlinear model of a synchronous machine connected to an infinite bus is given by Matthews et al. The system is shown in Figure 3.1.

The dominant dynamics of the nonlinear single machine can be simplified using the following assumptions:

- The voltage behind the transient reactance of the machine is constant.
- Governor/turbine dynamics are represented by a slow first-order system
- Swing equations are used to describe the mechanical motion of the synchronous machine.

The dynamics of the system are described by the following equations:

$$\dot{\delta} = \omega \tag{3.1}$$

$$\dot{\omega} = \frac{\omega_B}{2H}[P_m - P_{ac} - KP_{dc}] - D.\omega \tag{3.2}$$

$$P_{dc} = (\cos(\beta) - R_c I_d) I_d \tag{3.3}$$

$$\dot{I}_d = \frac{1}{L}(\cos(\beta) - R_c I_d) \tag{3.4}$$

$$\dot{P_m} = -\alpha P_m + v \tag{3.5}$$

where

δ: rotor angle of the machine in electrical radians relative to the center of mass.

ω: rotor angular velocity in radians per second with respect to synchronous speed.

H: inertia constant in seconds.

D: damping coefficient in seconds^{-1}

P_m: per unit mechanical power.

P_{ac}: per unit AC power.

P_{dc}: per unit power stored in the converter.

$\omega_B = 377$ rad/s

$\omega_b = 75.399$ rad/s

$K = 1$

α: time constant of governor/turbine or mechanical power actuator.

v: the corresponding input.

I_d: Direct current through converter.

R_c: Commutating resistance per unit.

$$X = X_d + X_t + X_l \tag{3.6}$$

$$P_{ac} = (E_1 E_2 / X) \sin \delta \tag{3.7}$$

Based on the dynamic model above, the states are defined as follows:

$x_1 = \delta$

$x_2 = I_d$

$x_3 = \omega$

$x_4 = P_m$

And the control inputs are:

$u_1 = \cos(\beta)$ and

$u_2 = v$

The system can be represented in state-space form as follows:

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \\ \dot{x_4} \end{bmatrix} = \begin{bmatrix} x_3 \\ -k_1 x_2 \\ -k_2 sin(x_1) + k_3 x_2^2 - D x_3 + k_5 x_4 \\ \alpha x_4 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ k_4 & 0 \\ -k_5 x_2 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

where,

$k_1 = \frac{R_c}{L}$, $k_2 = \frac{\omega_B E_1 E_2}{2HX}$, $k_3 = \frac{\omega_b R_c}{2HX}$, $k_4 = \frac{1}{L}$, $k_5 = \frac{\omega_B}{2H}$

The DC Converter is rated at 80 MW. The system is 230 kV and the machine rating is 800 MVA. On this rating base, the system parameters are [95]: $X = 0.2$ pu,

$R_c = 0.3$ pu, $L = 0.015$ pu, $H = 7.0$ s, $D = 0.5$ s^{-1}, and α = -0.1 s^{-1}.

This corresponds to $k_1 = 20$, $k_2 = 177.72857$, $k_3 = 8.078571$, $k_4 = 66.667$, and $k_5 = 26.928561$.

The control inputs are also constrained as follows:

$$-0.95 \leq u_1 \leq 0.985 \tag{3.8}$$

$$|u_2| \leq 3.5 \tag{3.9}$$

Due to the rating of the converter, limit is also imposed on x_2 (I_d) as: $0 \leq x_2 \leq 0.1$ pu. And since $x_4 = P_m$, it is required that $x_4 \geq 0$.

All of these constraints are easily dealt with in the implementation of MPC-PSO for this system as will be explained later.

3.3 MPC-PSO Control of SMIB System

In this section, we will study the application of the MPC-PSO to the SMIB system and comment on the results.

The primary control objective is to drive the machine from a perturbed, possibly unstable, state to a desired equilibrium point and to maintain it there.

A secondary control objective involves inducing fault conditions in the system while it is at equilibrium. They will disturb the equilibrium and the objective will be to bring the system back to the desired equilibrium points using the control inputs and maintain it there, in the presence or absence of the induced faults.

The control objectives involve these subgoals:

1. The machine must be operated at the rated frequency, i.e. change in frequency, x_3 must be zero at equilibrium.

2. The DC current through the converter, I_d, x_2 must be zero at equilibrium

3. A specified amount of AC power is required to be delivered to the bus. This defines the desired load angle, γ, of x_1.

Both primary and secondary control objectives are studied in this section and various cases of each are discussed.

3.3.1 Performance of Proposed Controller for Perturbed System

To achieve the primary control objective, the design procedure proposed in [95] is sliding VSC and it involves the following:

1. Transforming the state space system into a Luenberg canonical form.

2. Constructing a suitable sliding surface.

This procedure, especially the first step, is complicated and involves many manipulations. Furthermore, it will sacrifice the precision of control.

As in previous works, here the control is constrained as per the conditions in Section 3.2 with the following initial conditions:

$$x_1 = 0.0522, \quad x_2 = 0.1, \quad x_3 = 0.1, \quad \& \quad x_4 = 6.6\sin(x_1(0)) = 0.3444 \qquad (3.10)$$

There is no constant disturbance on the system in this case, but the initial states x_2 and x_3 are perturbed. This means that the system starts from a disturbed value and the aim is to bring these two perturbed states to zero.

The states of the system are:

$$X = [\delta \quad I_d \quad \omega \quad P_m] \tag{3.11}$$

And the control objective is to bring x_2 (I_d) and x_3 (ω) to 0 using $u_1 = \cos(\beta)$ and $u_2 = v$.

The cost function proposed is the following:

$$J = \sum_{i=1}^{H_p} \Delta I_d^2 + \Delta\omega^2 \tag{3.12}$$

where ΔI_d is the error in the DC current through the converter, I_d and $\Delta\omega$ is the error in the rotor angular velocity, ω in rad s^{-1} with respect to the synchronous speed of the rotor.

Case I - Control with Unconstrained Rate of Change on Inputs

In this case, the controls u_1 and u_2 are constrained within the limits defined in Equations 3.8 and 3.9. However, the control effort can change by any value within these limits, thus implying that there is no constraint on the rate of change of the control inputs Δu, i.e. Δu_1 and Δu_2 are unconstrained. The system is initialized from the perturbed initial states given in Equation 3.10 and the objective is to bring the con-

vertor current and change in frequency to zero as quickly as possible, while respecting the implied input constraints.

The controller succeeds in bringing the system from the perturbed system state to the steady state very quickly as seen in Figures 3.2 to 3.5. The behavior of the controlled outputs, I_d and ω are seen in Figure 3.3 and Figure 3.4 respectively. The convertor current, I_d, state x_2 goes to its steady state extremely quickly, taking only 0.03 seconds to reach the required equilibrium state of 0. More importantly, the change in frequency also becomes 0 and reaches the required equilibrium state after 0.1s. Practically, this means that the system frequency is brought to 60Hz after starting from an error of 0.1 p.u. The other states of the system, $x_1 = \delta$ and $x_4 = P_m$ settle at slightly different equilibrium points from the initial values after the perturbed system is brought to equilibrium. The new equilibrium value for x_1 (δ) is 0.0560 rad, up from the initial value of 0.0522, while that for x_4 (P_m) is 0.3694 p.u. up from 0.3444. Therefore, it is seen that due to the perturbation in initial conditions, the value of the states x_1 and x_4 change slightly.

The control effort applied is seen in Figures 3.6 and 3.7. The first control input, $\cos(\beta)$ is needed for only 0.03s. After that, I_d settles to zero. The second control effort, v is in effect for only 0.1s, the duration it takes for the frequency deviation to be zero. So, in total, the control efforts take only 0.1s to bring the system from perturbed state to equilibrium state, after which the control efforts attain an equilibrium value of zero. The control efforts, u_1 and u_2 are also found to be within the constraints imposed by the system in Equations 3.8 and 3.9. Since Δu is not constrained, the

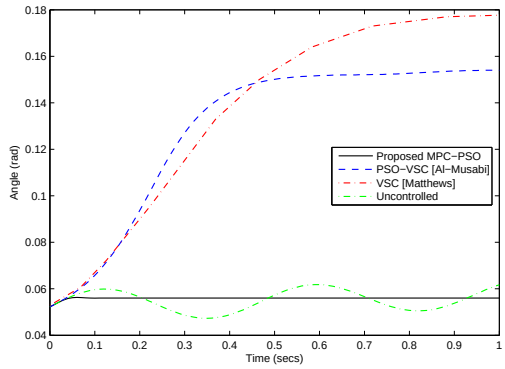

Figure 3.2: Angle (rad) for Case I

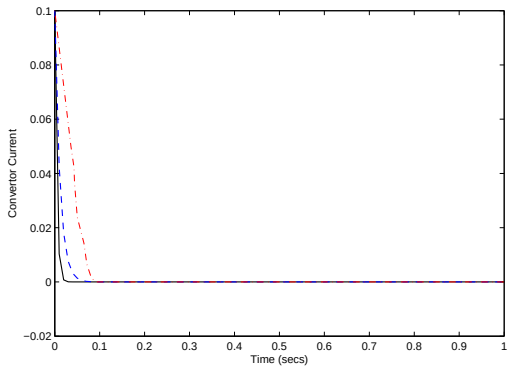

Figure 3.3: Convertor Current

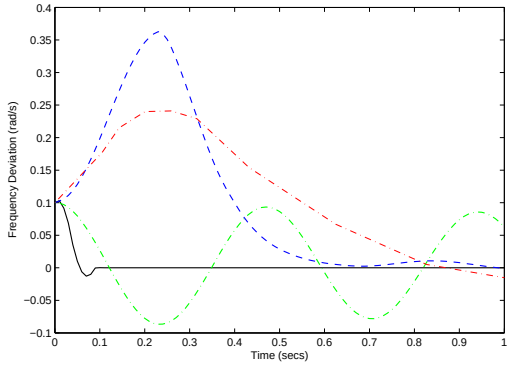

Figure 3.4: Frequency Deviation (rad/sec)

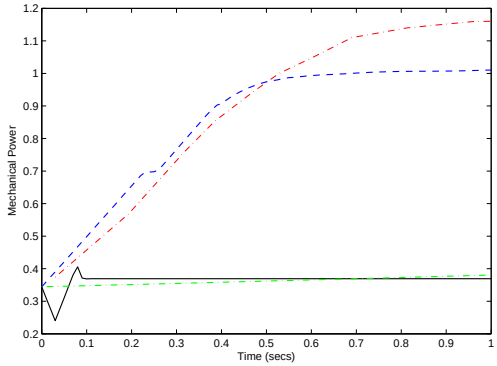

Figure 3.5: Mechanical Power

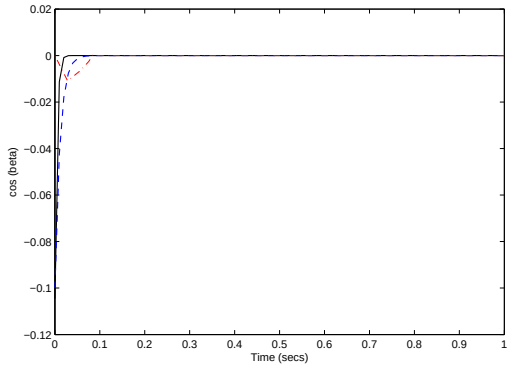

Figure 3.6: Control Input 1, $\cos(\beta)$

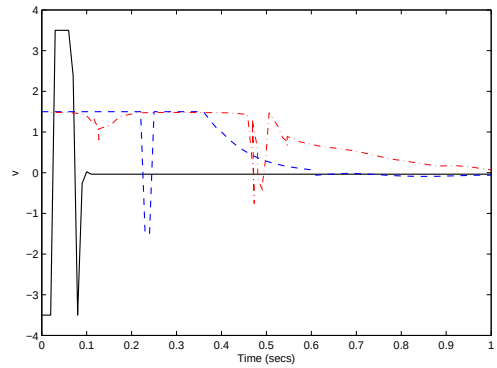

Figure 3.7: Control Input 2, v

controller rapidly changes the control inputs to the system. Both positive and negative control efforts are applied to the maximum in several instances and this is the reason the system arrives at equilibrium so quickly.

The Figures 3.2 to 3.5 also show the comparison of these results with Al-Musabi's [4] and Matthew's [95] work. It is seen that the proposed controller gives a massive improvement in results than the previously proposed Heuristics (GA/PSO/TS) based VSC (Al-Musabi [4]) and regular VSC (Matthews [95]). The proposed controller excels by bringing the system to the equilibrium states much quickly and keeping the deviation in the angle and mechanical power of the system minimal. The frequency deviation is found to be controlled in 0.1s, compared to at least 1s for previous works. This is a 10-fold improvement. The frequency deviation reached a maximum value of of only -0.013 p.u. while for the previous work, the deviation reached a maximum of 0.35p.u. at 0.25s. For the uncontrolled case in the figures, the system is of course unable to reach the equilibrium points.

The convertor current, I_d is also seen to reach the required equilibrium state in a shorter duration. Another important thing to be taken from this comparison is that it is seen that the proposed controller does not cause a large change in the angle and mechanical power of the SMIB system while bringing the perturbed frequency and current to equilibrium. So, it is seen that the angle is at equilibrium state is only 0.056 rad compared to at least 0.15 rad in the previous work [4]. Same is true for the mechanical power, which is only 0.369 p.u. compared to at least 1 p.u. in the previous work [4].

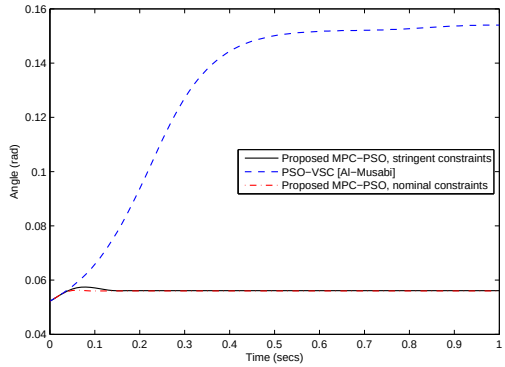

Figure 3.8: Angle for $|u| \leq \pm 1.5$

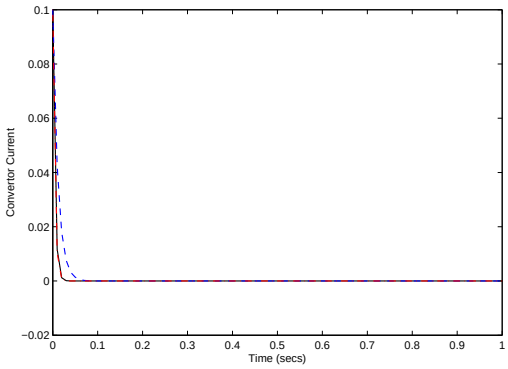

Figure 3.9: Convertor Current

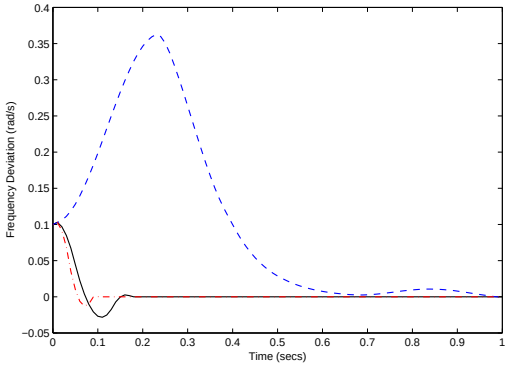

Figure 3.10: Frequency Deviation (rad/s)

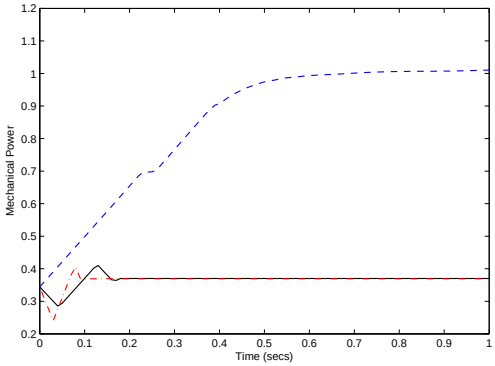

Figure 3.11: Mechanical Power

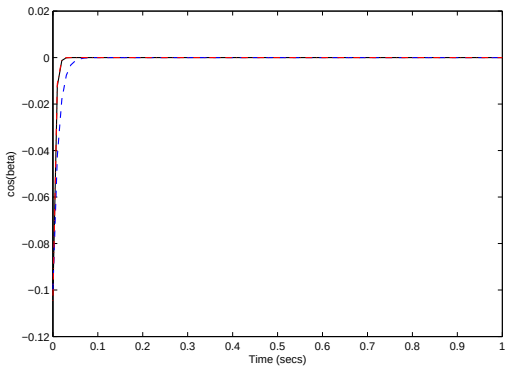

Figure 3.12: Control Input 1, $\cos(\beta)$

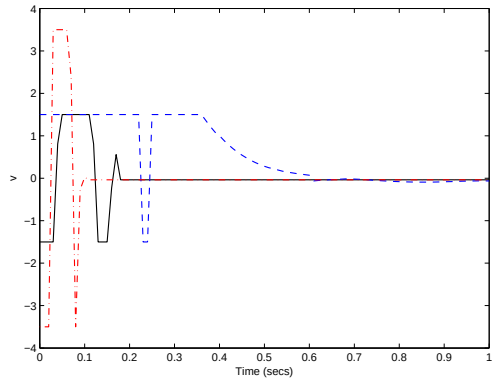

Figure 3.13: Control Input 2, v

It is also seen that the control effort, v is in effect for at least 1s in previous work [4]. However, during the whole duration, it is unable to reach the maximum allowable control limits defined in Equation 3.9. The maximum value it attains is ± 1.5. Using the proposed controller, the whole range of control input is utilized and the control input does reach the maximum allowable values of ± 3.5. This is the reason why the proposed controller gives the best results, by applying maximum allowable optimal controls to the system thus enabling it to reach the equilibrium states quickly.

For the sake of further comparison, we now extend this case by limiting the control effort v between ± 1.5. This is termed as an stringent constraint on v. Therefore,

$$-1.5 \leq u_2 \leq 1.5$$

The results for this case are seen in Figures 3.8 to 3.13 where they are compared with Al-Musabi's work as well as the proposed work here with nominal constraints on v. It is seen that now the value of $u_2 = v$ cannot go beyond ± 1.5. The results are of course better for the case when v could go beyond it and up to ± 3.5. However, it is demonstrated here that the results are still enormously better than the Al-Musabi's proposed PSO-based VSC. Also, the states x_1 and x_4 of the system are able to retain their equilibrium values in this case as well after going through minimal fluctuations. The frequency deviation is brought to the equilibrium value of 0 in 0.2s while the convertor current deviation becomes equilibrium at 0.03s. These results show that the proposed MPC-PSO controller is able to apply the right optimal efforts at the right time to bring the system to the required set points as quickly and as efficiently

as possible while obeying the constraints it is subjected to.

Case II - Control with Constrained Inputs

Now the case where constraints are imposed on the change of control efforts is considered. Therefore, a Δu is specified, so that the system changes the control effort only up to a certain value between each sample.

$$|\Delta u_1| \leq 1 \tag{3.13}$$

$$|\Delta u_2| \leq 1 \tag{3.14}$$

If the controller tries to switch the control effort by more than 1 between adjacent samples, it will be not allowed. The limit of 1 is mainly challenging for $u_2 = v$ as it is the more rapidly changing control, directly effecting the most critical output of change in frequency, ω. A limited Δu_1 doesn't have a profound effect as the state I_d comes to equilibrium very quicky in any case. Such a case, in which the controller is proposed with a constrained Δu is not found in the existing literature, so the results are compared with the nominal case of unconstrained Δu.

The behavior of the states in this case are seen in Figures 3.14 to 3.19. The controller takes a bit more than 0.2s to bring both the desired states x_2 & x_3 to steady state values, i.e. zero. The frequency deviation starts to decrease from the initialized value of 0.1p.u. and reaches an undershoot of -0.05p.u. at 0.1s the becomes zero at 0.23s. There are slight oscillations in I_d in this case also which continue up till 0.18s.

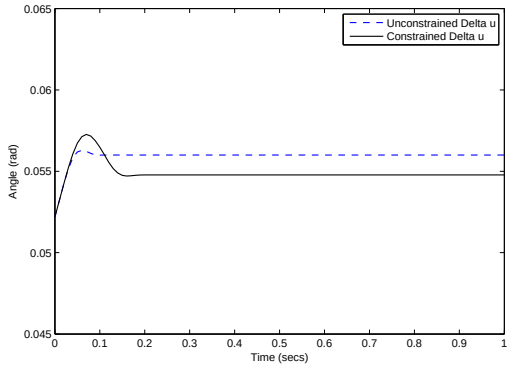

Figure 3.14: Angle (rad) for Case II

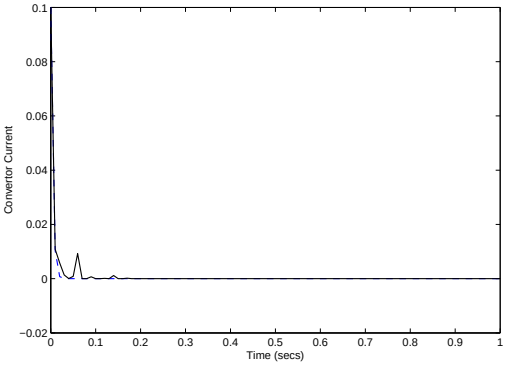

Figure 3.15: Convertor Current

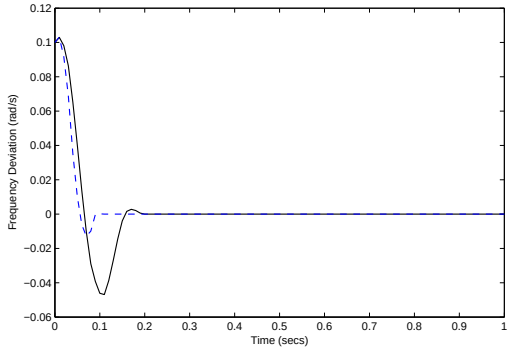

Figure 3.16: Frequency Deviation (rad/sec)

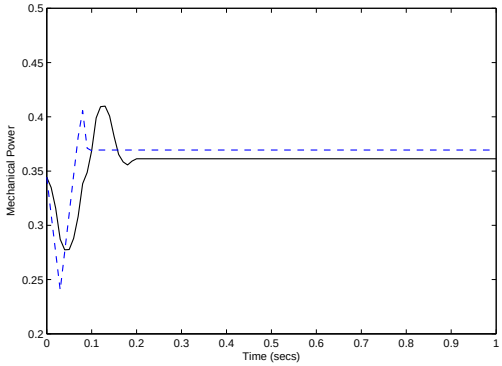

Figure 3.17: Mechanical Power

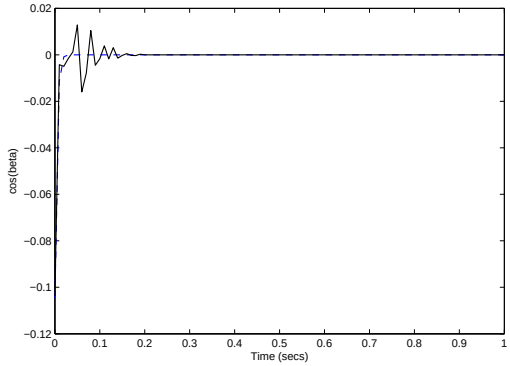

Figure 3.18: Control Input 1, $\cos(\beta)$

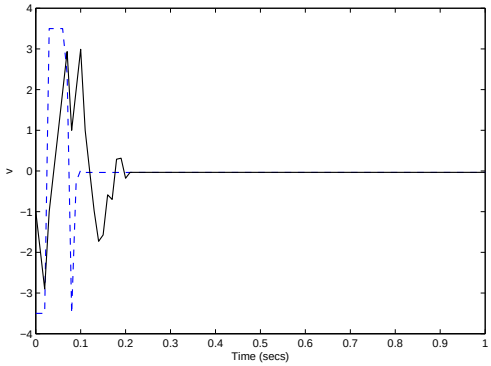

Figure 3.19: Control Input 2, v

The plots of control efforts are seen in Figures 3.18 and 3.19.The results confirm that both the constraints on u (Equations 3.8 and 3.9) as well as Δu (Equations 3.13 and 3.14) are not violated. Naturally, due to this restriction on the control effort it takes longer for the system to stabilize. The maximum u_1 applied is 0.018 and the maximum u_2 applied is 3 at two instances.

It is seen that the angle and mechanical power attain slightly different values because of the longer settling time now taken by the system. Comparing this with the previous work, even the constrained Δu controller is giving better results versus the unconstrained Δu VSC techniques.

3.3.2 Performance of Proposed Controller Under Fault Conditions

In this section, the control of SMIB system is presented in the event of real system faults, which are introduced in the lumped system components X, R_c and L. Cases like these have rarely been studied in the literature. Therefore, it will be interesting to study the performance of the newly proposed MPC-PSO controller in this way.

Case III - X changes by 10% Temporarily, $|\Delta u| \leq 1$

The collective reactance for the SMIB model is given in Equation 3.6. Due to a fault in one of the reactive components, the value of X changes by 10%, from 0.2 to 0.18. This fault appears for 10 cycles, from 0.2s to 0.3s, after which it is cleared. The control efforts are constrained as defined by the system, and the change in control effort is

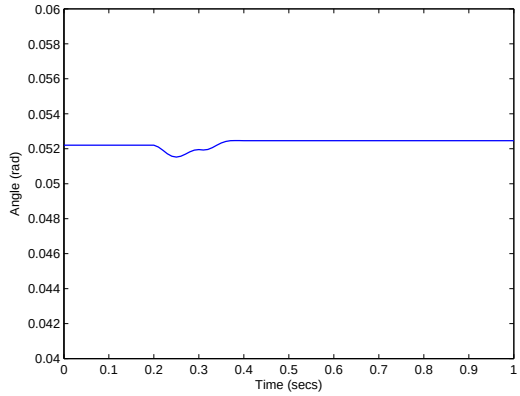

Figure 3.20: Angle (rad) for Case III

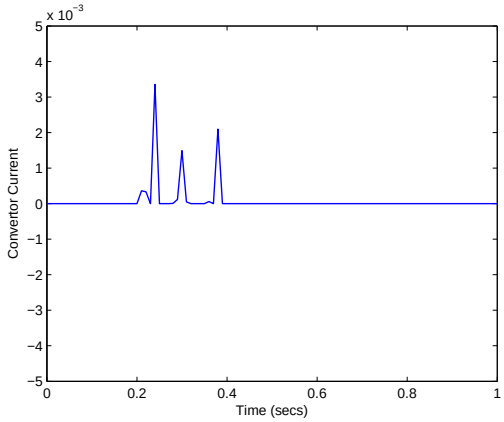

Figure 3.21: Convertor Current

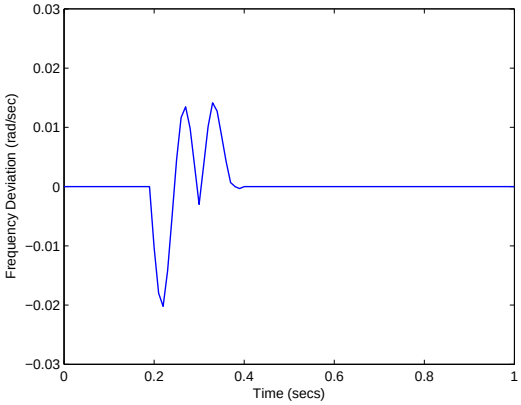

Figure 3.22: Frequency Deviation (rad/sec)

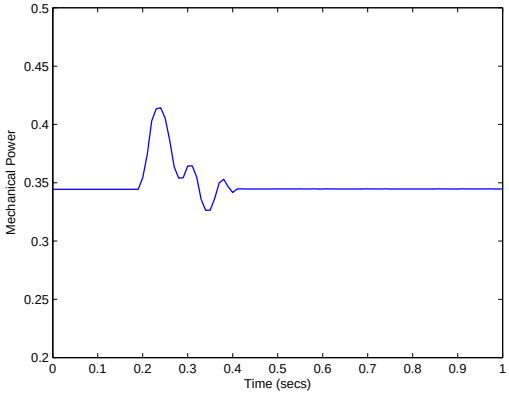

Figure 3.23: Mechanical Power

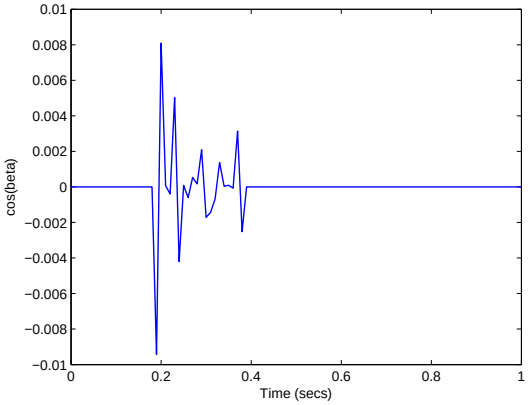

Figure 3.24: Control Input 1, $\cos(\beta)$

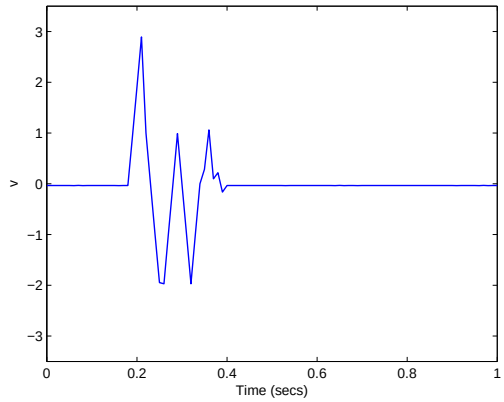

Figure 3.25: Control Input 2, v

constrained by $|\Delta u| \leq 1$.

The reaction of the system states to this fluctuation in the value of X is given in Figures 3.20 to 3.23. It is seen that once the fault appears, the system frequency deviates sharply reaching a maximum value of -0.02 rad/s at 0.22s. The frequency deviates around the nominal value of 60Hz several times as evident by the oscillations in Figure 3.22. This is because the fault disappears at 0.3s and thus there is further fluctuation in the frequency, which reaches 0.024 rad/s at 0.34s. The system then reaches the equilibrium condition again at 0.46s, that is, 0.26s after appearance of the temporary fault. The convertor current varies by maximum values of -0.005 and +0.005 p.u. and reaches equilibrium again at 0.46s. So both the disturbed system states are controlled in less than 0.3s and brought back to equilibrium states. The angle and mechanical power of the system deviate by very small amounts due to the appearance of this fault.

Figures 3.24 and 3.25 show the control inputs for this case. Δu is constrained in this case. Of course, the system is also obeying the limits on u_2 described in Equations 3.8 and 3.9. The maximum value for u_1 is 0.008 while for u_2, it is ± 2.8. Both control efforts attain their steady state values once the system reaches equilibrium at 0.46s.

Case IV - X changes by 10% Permanently, $|\Delta u| \leq 1$

This case is similar to the previous one, however now the value of X changes by 10%, from 0.2 to 0.18 at the instant 0.2s and remains so. The constraints on u and Δu are same as in Case I.

Due to the fault, both the controlled states of the system deviate from the steady

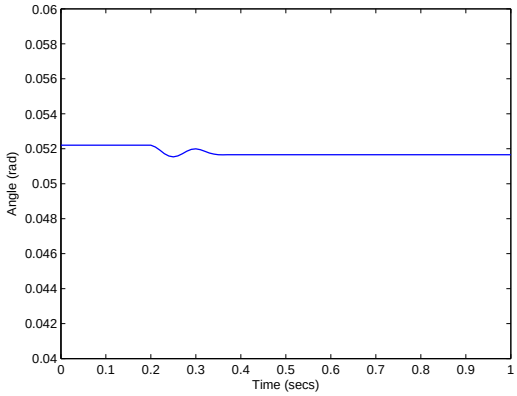

Figure 3.26: Angle (rad) for Case IV

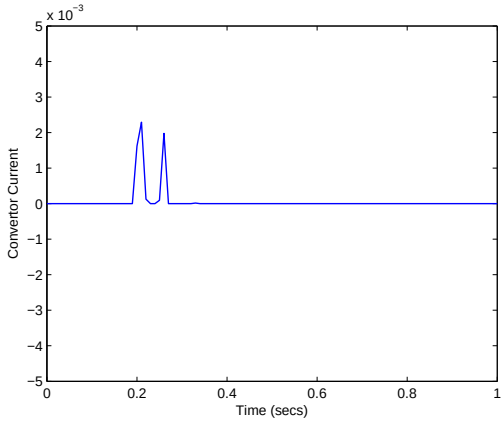

Figure 3.27: Convertor Current

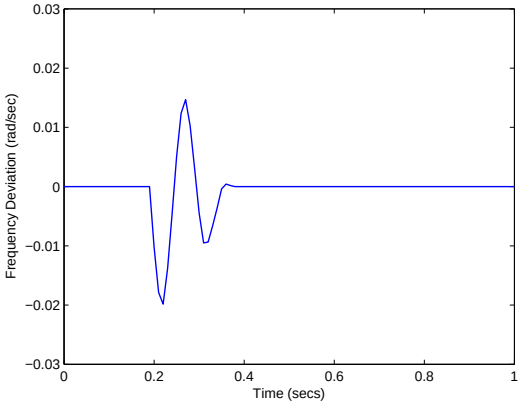

Figure 3.28: Frequency Deviation (rad/sec)

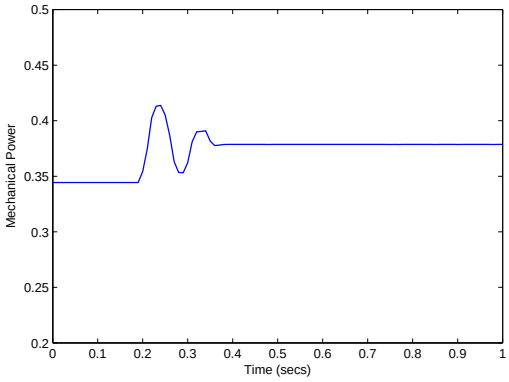

Figure 3.29: Mechanical Power

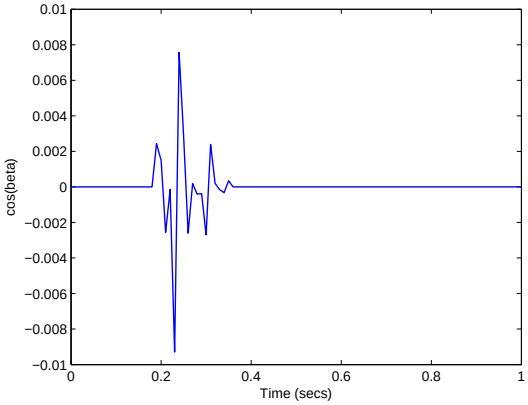

Figure 3.30: Control Input 1, $\cos(\beta)$

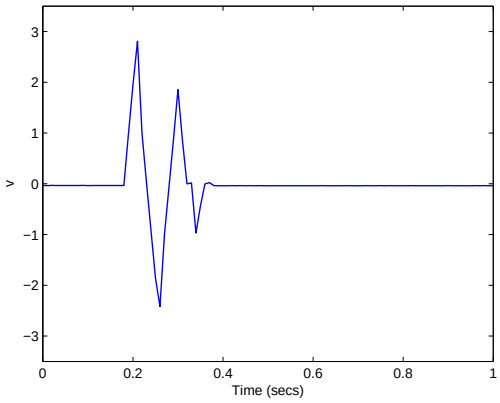

Figure 3.31: Control Input 2, v

state values. The current, I_d changes by as much as 0.005p.u. and the frequency deviates by a maximum of -0.02 rad/s. The controller, although constrained, brings the controlled states to steady values within 0.2s of the fault. The results for this case are given in Figures 3.26 to 3.31.

It is evident from these results that the control signals applied to the system obey all constraints. The values of u_1 and u_2 remain within the specified constraints, as well as $|\Delta u_1|$ & $|\Delta u_2|$ are always ≤ 1.

In this case, it is observed that the system reaches equilibrium quicker than the previous case because the fault in X is not fluctuating. In fact, it is a parameter variation the controller copes up with this by applying continuous optimal control inputs. Therefore, due to this variation, it is observed in Figures 3.26 and 3.29 that the states of the system angle and mechanical power change.

Case V - Parameter Variations in the System

In this extreme case, the system with unconstrained Δu has a parameter variation given by the following:

1. The value of R_c changes by 50%, from 0.3 to 0.6.

2. The value of L changes by 50%, from 0.015 to 0.03.

3. The value of X changes by 50%, from 0.2 to 0.1.

The response of the system in this case is seen in Figure 3.32 to 3.35. The frequency deviates by as much as -0.132 p.u. while the current fluctuate by a maximum of 0.204 p.u., however, these states are controlled back at their steady values in less than 0.2s.

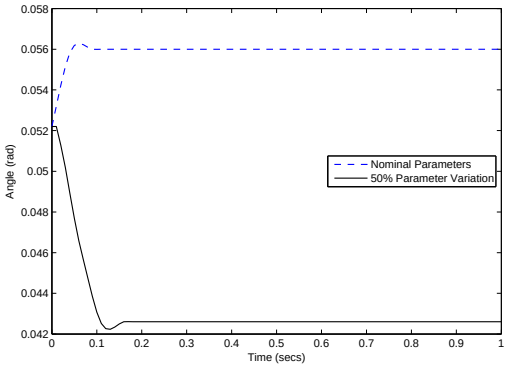

Figure 3.32: Angle (rad) for Case V

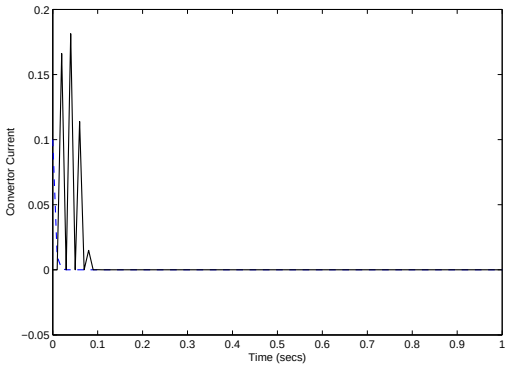

Figure 3.33: Convertor Current

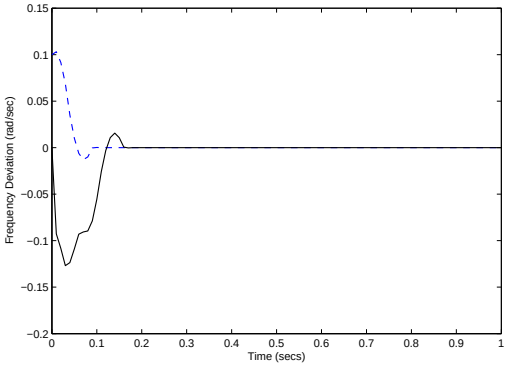

Figure 3.34: Frequency Deviation (rad/sec)

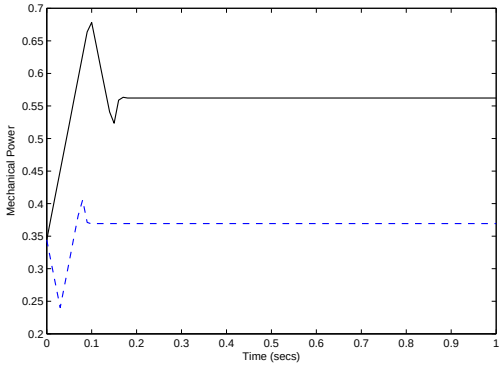

Figure 3.35: Mechanical Power

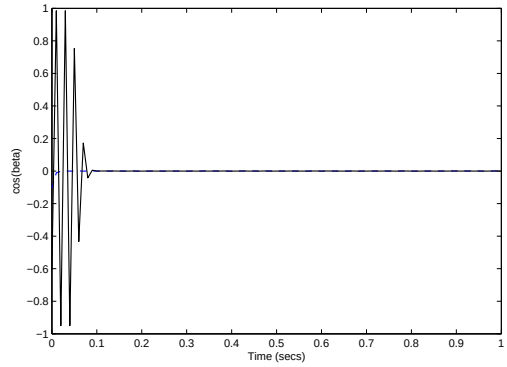

Figure 3.36: Control Input 1, $\cos(\beta)$

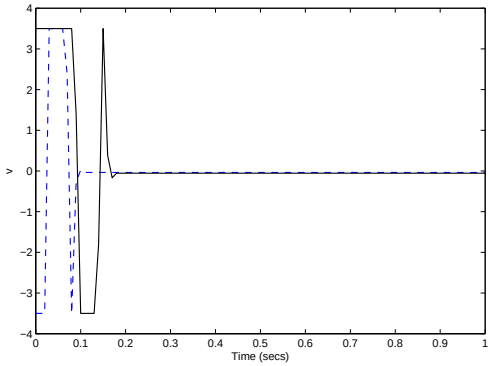

Figure 3.37: Control Input 2, v

It is good to see that the system is able to work at steady state even in the presence of major parameter variations which appear in all resistive and reactive components. The control efforts applied in this case are seen in Figures 3.36 and 3.37. The controller applies large control efforts at several time instants to bring the system to steady state as quickly as possible, however, the constraints defined by Equations 3.8 and 3.9 are always met. The controller is able to control the system frequency and convertor current to equilibrium states in less than 0.2s. However, due to the change in system parameters, the equilibrium values of angle and mechanical power, that is, states x_1 and x_4 shift. This shift is compared in the figures which show the behavior of the system with nominal parameters vs. 50% parameter variation.

3.4 Conclusion

The predictive controller based on PSO applied to a SMIB model performed very well. Results obtained for the cases of perturbed initial conditions as well as in the event of induced faults showed that MPC-PSO can effectively control the SMIB system. The overshoots in the system states as a result of these perturbations and faults were minimal and the control effort applied was smooth and quick. The controller, although challenged by system as well as control constraints and being limited by constraints on change in control effort, performed well in controlling the complex, nonlinear, MIMO system directly without any need of linearization, model reduction of other approximations. This shows the ability of the MPC-PSO controller to be applied directly to complex power systems with the promise of expecting good results. A comparison of

the results obtained with preceding literature showed that predictive control performs far better than variable structured control in this case. The PSO algorithm incorporated in the MPC technique enabled the controller to keep the computational time low and arriving at optimal solutions of the control effort.

CHAPTER 4

APPLICATION OF MPC-PSO TO LOAD FREQUENCY CONTROL

4.1 Introduction

Load Frequency Control (LFC) has been one of the most important subjects for power systems engineers for decades as it is essential to maintain the supply of high quality and reliable electric power to the consumers. It is also known as Automatic Generation Control (AGC) [27]. The main objectives of LFC for a power system are [131]:

- Ensuring zero steady-state errors for frequency deviations.
- Minimizing unscheduled tie line power flows between neighboring control areas.
- Getting good tracking for load demands and disturbances.
- Maintaining acceptable overshoot and settling time on the frequency and tie line power deviations.

Today's large scale power systems are composed of interconnected subsystems or control areas in which the frequency of the generated power has to be kept constant. These subsystems are connected via tie-lines or HVDC links making distinct control areas [176]. Each area has one or more generators and is responsible for its own loads as well as scheduled interchanges with neighboring areas. However, loading in power systems is never constant and changes in load result in changes in system frequency.

Interconnected neighboring areas in a modern power system pose a challenge to load frequency control which must be addressed to maintain the quality of the whole power system. The main quality risk involved is that control area frequencies as well as tie-line power can undergo prolonged fluctuations due to a sudden change of loading in an adjacent power system. These prolonged fluctuations are mainly the result of system nonlinearities.

LFC has undergone extensive investigation because of its importance. The need of regulating the output power (and voltage) of each generator of the control area is very important while maintaining the frequency fluctuations within pre-specified limits. The goal is to minimize frequency deviations all interconnected areas as well as in the tie-lines.

The LFC problem has been studied for almost four decades. A large portion of the study has considered linear LFC problems only. One of the earliest studies is by Cavin, which considers the LFC problem from an optimal stochastic control point of view [26]. The application of this technique resulted in improved transient response of the power and frequency deviations. However, this required the implementation of

a fifth-order filter and was quite complex. A simpler technique based on PI optimal regulator is given in [21]. Other methods of classical control were also applied to LFC. However, with these methods, the dynamic performance was poor, especially with nonlinearities or parameter variations.

Adaptive control has been applied to LFC extensively and suboptimal control techniques have also been developed due to practical limitations of the optimal techniques [137], [33]. One of the early adaptive control methods is the PI adaptation technique given by Pan and Liaw [104]. It considers the plant parameter changes and instead of using an explicit parameter identification, the controller only used the available information of states and outputs fed back to it. Good results were obtained even with this reduced order plant model and the performance was somewhat insensitive to parameter variations and generation rate constant nonlinearity. Liaw has also presented a reduced order adaptive LFC technique for interconnected hydrothermal power system [89]. An adaptive decentralize LFC scheme for multi-area power systems is given by Zribi et al that guarantees very small fluctuations [176]. A similar technique is also given by Bevrani [14].

Recently, fuzzy logic has also been extensively applied to the LFC problem. The fuzzy techniques perform well to remove steady state errors and are simpler and easily applicable [28], [22]. However, they have poor transient response, and to improve this fuzzy PID methods have been proposed [167]. Fuzzy systems have also been combined with ANN, GA and other techniques. Shayeghi et al have provided a PSO based multi-stage LFC technique in [129]. A useful comparison of intelligent techniques used on

linear LFC is given by Mathur [94].

Another important technique applied to LFC has been the use of Variable Structure Control (VSC). Several papers are available on this topic [138], [86], [85], [38]. These VSC schemes, however, used trial and error methods to select LFC feed back gains. This is a cumbersome and inaccurate process and can be computationally impractical. Therefore, heuristic techniques have been developed to select the feed back gains for VSC. A good example is in the paper by Al-Hamouz and Al-Duwaish, which uses GA to select the optimal VSC feedback gains [3]. The results show improved dynamic performance of the system with much less control efforts being used. Other schemes using GA have been also promoted in the literature, for example, the GA and LMI based Robust LFC given in [119].

More recently, Model Predictive Control (MPC) has also been applied to the LFC problem in the form of State Contractive Constraint (SCC) based MPC [83]. The scheme was applied to a two area linear LFC system and it guaranteed stability with uncertainties in the model. However, the control was not smooth and had quite a bit of oscillations.

Although good linear control of multiarea load frequency has been achieved by several researchers, these designs will not work properly in practice due to the real nonlinear nature of LFC systems. Therefore, consideration of nonlinearities in the models of LFC is very important. One of the main type of nonlinearities is the Generation Rate Constraint (GRC). This is the constraint on the power generation rate of the turbine and due to it the disturbance in one area affects the output frequency

in other interconnected areas. The load frequency in the tie-line is also affected. The Governor Dead Band (GDB) is also another type of nonlinearity in the LFC systems [153].

For nonlinear system models of LFC, a Ricatti-based optimal control technique is proposed by Wang [155]. The controller design is based on optimization of a Ricatti-equation. The results show large variations in frequency and power output and the system states take a long time to settle to steady state values. Using the proposed technique, these important problems will be tackled.

Adaptive control provides a better control of the LFC problem, especially with the presence of nonlinearities and parameter variations [123], [149]. However, implementation limitations hamper its popularity. Decentralized load frequency control designs have been given by Yang et al, [165], [166] using LQR techniques. This gives improved results over the Ricatti-based optimal control techniques and the aim of the proposed technique is to improve that further.

Velusami gives decentralized biased dual mode controllers with for LFC nonlinearities which are designed on the basis of ISE and stability criteria. Its implementation is easy due to the simple structure of the dual mode controller. The results showed good closed loop stability with high quality responses of the system for both steady and transient states while being less sensitive to parameter variations. Thus it appears that adaptive and decentralized control techniques for LFC give better results compared to conventional schemes. A detailed discussion of different control techniques used on LFC and their performance is given in [131].

For the past decade, researchers have focused on intelligent control schemes for nonlinear LFC as well. Birch gives an enhanced neural network LFC technique for the power system in England and Wales [15]. The NN approach has several advantages of conventional approaches, as the controller is able to perform well in case of parameter variations and time variance of the system, resulting in effective and robust control. However, the drawback is that the bulky neural network has to be trained offline and is not suitable for full closed loop control. It also has to be retrained in the case of system changes. and Hameida [59]. Shayeghi has also given an H∞ based robust ANN LFC scheme [130]. Other ANN based techniques are given in good detail in the survey [131].

A PSO technique for VSC of the nonlinear LFC problem is presented in [4]. The use of PSO improves the calculation time of the results drastically, while also improving the dynamic performance of the LFC problem. The results show improvements over optimal integrated control, however the results using decentralized VSC technique [166] still fare better. The aim of the proposed technique is to improve the oscillatory transient response and achieve faster settling of system states. Furthermore, the PSO-VSC technique cannot sufficiently prevent frequency deviation in the adjacent interconnected areas. A PSO based VSC technique is also given in [55]. A new approach to LFC problem has been the use of Internal Model Control (IMC) to tune the PID controller [145].

It is concluded from this review that nonlinearities play a very important role in the quality of the LFC system and they cannot be ignored. Furthermore, the objective

is to achieve minimum frequency deviation in an LFC system when it is subjected to disturbance. Several of the techniques cited have been successful to achieve these results. However, none of the previous work on nonlinear LFC has handled the system constraints in the controller design process. The proposed approach will handle the normally present nonlinearities and constraints in the LFC system in a structured way in the controller design phase. This will give obvious advantages with regards to optimal control and constraints handling.

One of the reasons to opt for MPC as a load frequency controller is that MPC is now a well-known control technique that has proven reliability. Also, it has been shown that it can easily incorporate and handle nonlinearities and constraints. Although it has been extensively used to control industrial processes successfully, it has found very limited use in the field of power systems. Therefore, it is aimed to be a new frontier for MPC. Moreover, the use of PSO in the proposed MPC technique gives more flexible, accurate and faster control of processes and power systems alike.

4.2 Model of an LFC System

The dynamic model for an n-area interconnected system is given in this section. The model is taken from the work done by Zribi [176] and Yang [166]. The model can be

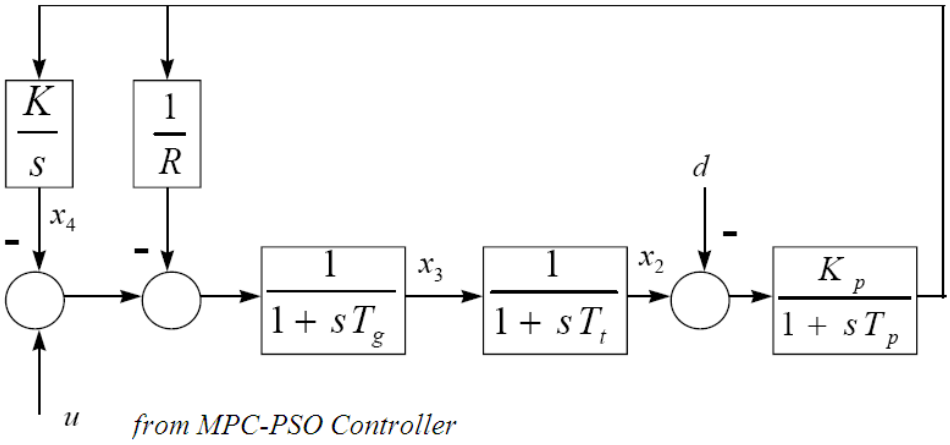

Figure 4.1: Block diagram of single area LFC

presented as the following:

$$\dot{X}_i(t) = A_i x_i(t) + B_i u_i(t) + \sum_{j=1,\, j\neq i}^{n} E_{ij} x_j(t) + F_i d_i(t) \tag{4.1}$$

$$y_i(t) = C_i(t) x_i(t) \tag{4.2}$$

$$Where, \tag{4.3}$$

$$\dot{X} = \begin{bmatrix} \Delta \dot{f}_i(t) \\ \Delta \dot{P}_{g_i}(t) \\ \Delta \dot{X}_{g_i}(t) \\ \Delta \dot{P}_{c_i}(t) \\ \Delta \dot{P}_{t_i}(t) \end{bmatrix} \tag{4.4}$$

$$A_i = \begin{bmatrix} -1/T_{p_i} & K_{p_i}/T_{p_i} & 0 & 0 & -K_{p_i}/T_{p_i} \\ 0 & -1/T_{t_i} & 1/T_{t_i} & 0 & 0 \\ -1/R_i T_{G_i} & 0 & -1/T_{G_i} & -1/T_{G_i} & 0 \\ K_{E_i} & 0 & 0 & 0 & K_{E_i} \\ \sum_j T_{ij} & 0 & 0 & 0 & 0 \end{bmatrix} \tag{4.5}$$

$$B_i^T = \begin{bmatrix} 0 & 0 & 1/T_{G_i} & 0 & 0 \end{bmatrix} \tag{4.6}$$

$$E_{ij} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -T_{ij} & 0 & 0 & 0 & 0 \end{bmatrix} \tag{4.7}$$

$$F_i^T = \begin{bmatrix} -K_{p_i}/T_{p_i} & 0 & 0 & 0 & 0 \end{bmatrix} \tag{4.8}$$

$$d_i(t) = P_{d_i}(t) \tag{4.9}$$

The definitions of the symbols used in the model are as follows:

$\Delta f_i(t)$: incremental change in frequency for ith area subsystem (Hz)

ω: rotor angular velocity in radians per second with respect to synchronous speed.

H: inertia constant in seconds.

D: damping coefficient in seconds^{-1}

P_m: per unit mechanical power.

P_{ac}: per unit AC power.

P_{dc}: per unit power stored in the converter.

$\omega_B = 377$ rad/s

$K = 1$

The control objective of LFC is to keep the change in frequency, $\Delta f_i(t) = x_1(t)$ as close to 0 as possible in the presence of load disturbance, $d_i(t)$ by the manipulation of the input, $u_i(t)$.

4.3 Single Area Load Frequency Control

In single area LFC, E_{ij} is ignored as there are no tie lines. Using the following system parameters [138]:

$T_p = 20$s, $K_p = 120$ Hz p.u. MW^{-1}, $T_t = 0.3$s, $K = 0.6$ p.u. $MW^{-1}\ rad^{-1}$, $T_g = 0.08$s, $R = 2.4$ Hz p.u. MW^{-1}

The corresponding values of A, B & F are:

$$A = \begin{bmatrix} -0.05 & 6 & 0 & 0 \\ 0 & -3.33 & 3.33 & 0 \\ -5.208 & 0 & -12.5 & -12.5 \\ 0.6 & 0 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 & 12.5 & 0 \end{bmatrix}^T$$

$$F = \begin{bmatrix} -6 & 0 & 0 & 0 \end{bmatrix}^T$$

The proposed MPC-PSO described in Chapter 2 has been applied on this system to minimize certain cost functions to obtain the optimal control signals. The objective is to keep the load frequency at the specified rate, or more specifically, to keep the change in load frequency equal to zero in the presence of disturbance at the loading.

The PSO parameters used for the controller are particles, $n = 20$, maximum number of iterations, $m = 500$, prediction horizon, $H_p = 7$, $w_{max} = 0.9$, $w_{min} = 0.4$, $c_1 = c_2 = 2.04$ and a time varying weight is used. The iterations ensure that the

algorithm is terminated after there is no possibility of significant improvement in the value of the cost function.

The cost function used is,

$$J = \sum_{i=1}^{H_p} e(k+i)^T Q e(k+i)$$

where $Q = 1$. The constraint on the control signal is:

$$-0.2 \leq u \leq 0.2$$

The system is simulated first for the linear case and then for nonlinear case with parameter variations. Initially, all states are at zero.

4.3.1 Single Area LFC Excluding Nonlinearity

First, Single Area LFC excluding the GRC nonlinearity is explored. The system is given a step change of 0.3 p.u., which means that the load of this generation system has changed by 0.3 p.u. Naturally, this change in load will demand the system to adjust its load by the same amount to power it. This will change the load frequency. The MPC-PSO based Load Frequency controller needs to minimize this frequency deviation as well as bring it to zero as soon as possible while obeying the constraints of the system as well as the control effort. The constraints on the system are defined in its model while the constraints on the control effort are designed into the controller.

The behavior of the system is studied under three conditions. In all these, the

limits of the control signal are imposed to be,

$$|u| \leq 0.2$$

These conditions are designed on the basis of constraint on the control effort as:

- Design 1 - No constraint on change in the control effort between samples, i.e. Δu is unconstrained.
- Design 2 - $\Delta u \leq 0.1$. This means that the control effort cannot change by more than 50% between samples.
- Design 3 - $\Delta u \leq 0.05$. This means that the control effort cannot change by more than 25% between samples.

Figure 4.2 shows the disturbance of 0.3 p.u. applied to the system and the corresponding frequency deviation observed. All the three designs are compared. It is observed that the disturbance causes the least frequency deviation for the case when the control effort is unconstrained between samples. The reason is obvious. The frequency deviates to a maximum value of -0.02 p.u. For the case of constrained Δu, the frequency deviation is relatively large, up to a value of 0.03 p.u. However, it is also observed that for the case of unconstrained Δu, the change in generated power is larger than the case with constrained Δu as shown in Figure 4.3. This means that to change the output power with respect to the load disturbance, there momentarily is an overshoot going up to 0.07 p.u. which for the case of constrained Δu is only up to 0.05. After this value, the generated power steadily drops to the required 0.03 p.u.

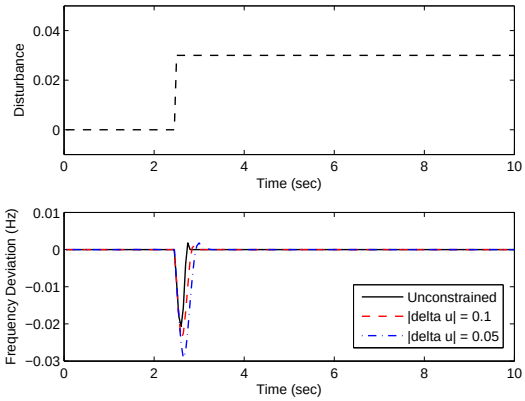

Figure 4.2: Disturbance and Frequency Deviation for Designs 1-3

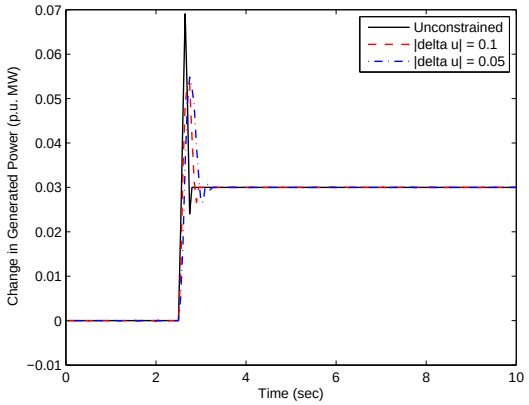

Figure 4.3: Change in Generated Power for Designs 1-3

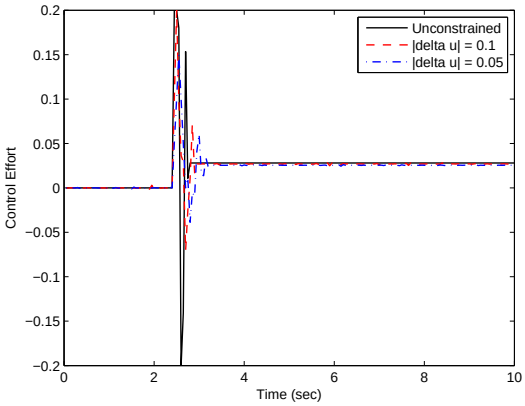

Figure 4.4: Control Effort for Designs 1-3

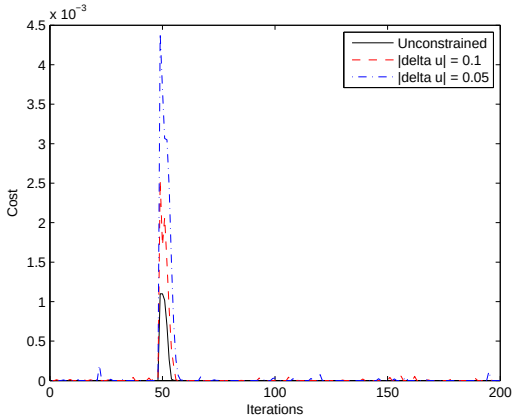

Figure 4.5: Cost Function for Designs 1-3

value. Same is true for the small undershoot. Understandably, it takes more time to achieve the results in the case of constrained Δu.

The control effort is shown in Figure 4.4. It is seen that the maximum control effort is constrained between -0.2 and 0.2 at all times, which were the original self imposed constraints on the system. For the case of unconstrained Δu, the control effort varies sharply and reaches both maximum and minimum allowed values. However, for the case of $\Delta u \leq 0.05$, the control effort reaches a maximum value of 0.13 and it is observed that it remains within the specified constraint between samples.

The cost function is shown in Figure 4.5. The cost function takes its maximum value when the disturbance just appears, based on which the correct control efforts are found and applied to the system resulting in the cost function to be minimized gradually. Naturally, for the controller designed with constrained Δu, the cost function attains the maximum value.

To study the robustness of the proposed controller for the case of varying load disturbances, a load disturbance seen in Figure 4.6 is applied. The load is simulated to vary from a disturbance of 0 p.u. to 0.03 p.u., going up to 0.05 p.u. and then becoming 0 p.u. again. This effect of the varying load is on the load frequency is also seen in this figure. It is seen that the load frequency varies most when the disturbance varies most. When the disturbance varies from 0.05 p.u. to 0, the load frequency varies maximum for the case of controller with constrained Δu, going up to a maximum frequency disturbance of 0.05 p.u. and 0.045 p.u. for the case of unconstrained Δu. The corresponding behavior of the change in generated power is

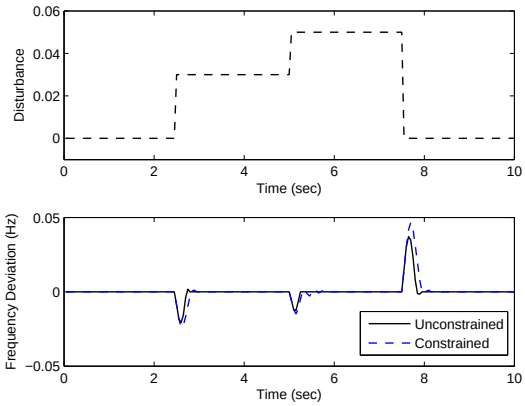

Figure 4.6: Robustness of Frequency Deviation

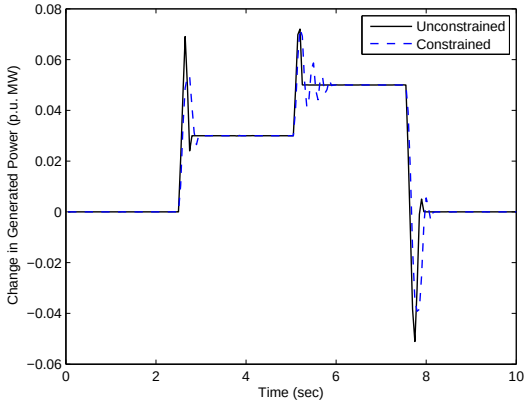

Figure 4.7: Robustness in Change in Generated Power

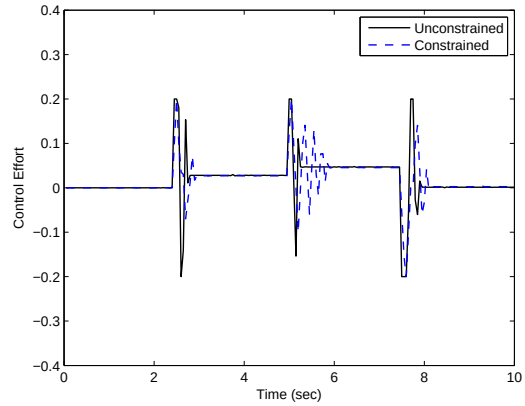

Figure 4.8: Control Efforts

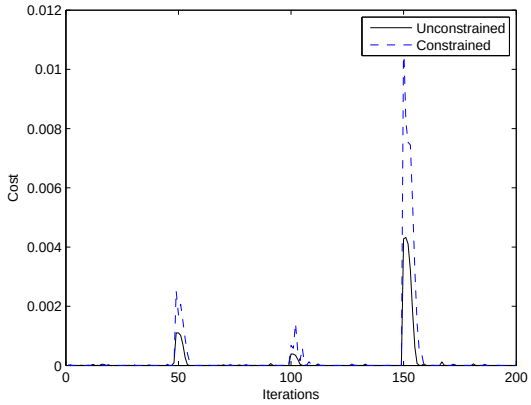

Figure 4.9: Cost Functions

seen in the Figure 4.7. It is seen that the change in generated power follows the load disturbance meaning that the system can supply the load its power demand. The power generated changes most when the disturbance is largest. Also, the trade off seen in the previous results is also apparent here, that the change in generated power is more for the case of unconstrained Δu, however the frequency deviation is large and vice versa for the case of constrained Δu. It is also seen that the change in generated power takes a few more instances to arrive at the steady state for the case of constrained Δu. This behavior is in line with the observations of the previous case as well.

The control effort for both cases is seen in Figure 4.8. It is seen that the controller always obeys the constraint $|u| \leq 0.2$. The cost function is seen in the Figure 4.9. It is seen that the cost if largest for the case when the disturbance is most. Understandably, it is more for the case when the controller has a constrained Δu than for the case of unconstrained Δu.

4.3.2 LFC Including GRC Nonlinearity

Now the nonlinearity in the form of Generation Rate Constraint (GRC) is added to the system. The nonlinearities appear in the form of saturation of states and can be illustrated by the general n-area block diagram in Figure 4.10.

A GRC value of 0.6 p.u. $MW\ min^{-1}$ = 0.01 p.u. $MW\ sec^{-1}$ is applied to the system. This means that the generated power output of the system cannot vary by more than 0.01 p.u. MW in 1 second. A disturbance of 0.01 p.u. is present in the

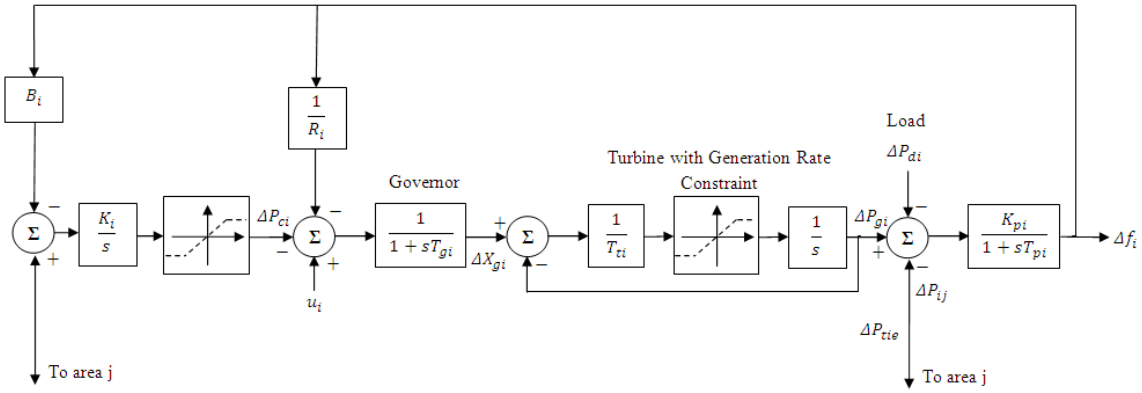

Figure 4.10: Block diagram of nth area LFC with GRC nonlinearities

system. The proposed controller is applied to the system with this nonlinearity. The control effort in this case is also restricted to be $|u| \leq 0.5$. The prediction horizon, $H_p = 3$. All other conditions are same as in the previous case.

The results for this case are seen in Figures 4.11 to 4.17. It is seen that for the case when GRC is present with a constant power demand of 0.01 p.u. from the system, the frequency deviates to a maximum value of -0.033 Hz as seen in Figure Figure 4.11. The frequency stops deviating after 6.3s. It can also be seen from Figure 4.13 that the generated power from the system cannot change beyond 0.01 p.u. MW in 1 second. The system is able to supply the required power demand after 4.15s, however, there are a few fluctuations after that, and the system is able to completely reach the steady state at 6.2s. The power generated from the system is now 0.01 p.u. MW. Compared with the PSO-VSC proposed by Al-Musabi [4], it is seen that the proposed controller offers significant improvements over the previous work as seen in Figure 4.11. The frequency deviation is much lesser and the load frequency deviation becomes zero quicker. It is also seen from Figure 4.13 that the power demand is fulfilled quicker

and more smoothly using the proposed technique. The control effort is seen in the Figure 4.15. The maximum control effort applied to the system is 0.14 and it is applied at the time when frequency deviation is maximum as clear from the figures. After the system reaches steady state with no frequency deviation with the required change in the generated power, the control effort is set at a constant value of -0.027. The cost is seen in Figure 4.17 and understandably, it is most at the start when the frequency deviation is maximum and then gradually decreases to become zero.

The system is also tested for a GRC value of 0.1 p.u. $MW\ min^{-1} = 0.0017$ p.u. $MW\ sec^{-1}$, as done in previous work [4], [155]. The results of this test, along with the comparison with the previous work can be seen in Figures 4.12 and 4.14. It is seen that the proposed technique preforms much better than that Riccati-based optimal load frequency controller proposed by Wang [155]. Comparing with the PSO-VSC technique given by Al-Musabi [4], the performance of the proposed technique is almost same. It can be seen in Figure 4.12, that the maximum frequency deviation of the system using the proposed technique is lesser than the previous work for this value of GRC. The control effort for this case is given in Figure 4.15, while the cost is given in Figure 4.17.

Another challenging test for the LFC system is through varying the load disturbance. A varying load disturbance, as seen in Figure 4.19 is applied to the single area system with GRC = 0.01 p.u. $MW\ sec^{-1}$. The load disturbance is 0.01 p.u. at the start and then changes to 0.02 and 0.03 p.u., and finally becomes 0.015 p.u. The dynamics of the frequency deviation and change in generated power are seen in Figures

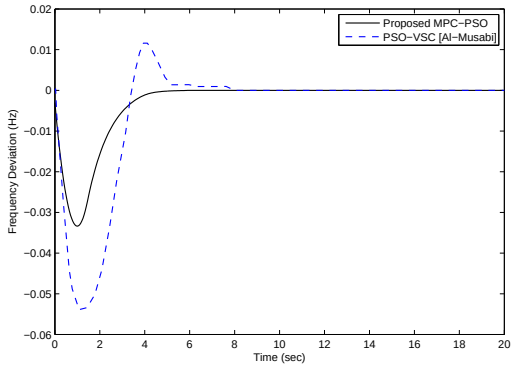

Figure 4.11: Frequency Deviation for GRC = 0.01

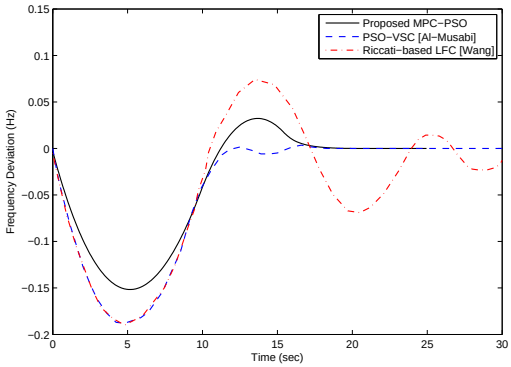

Figure 4.12: Frequency Deviation for GRC = 0.0017

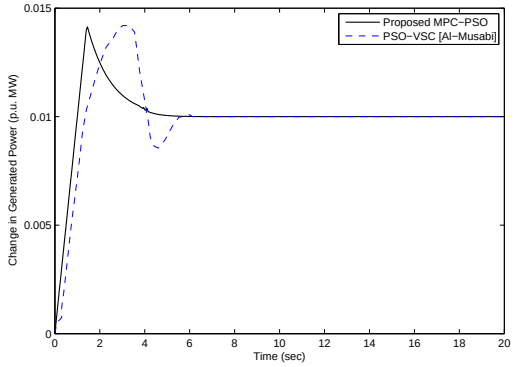

Figure 4.13: Generated Power Output for GRC = 0.01

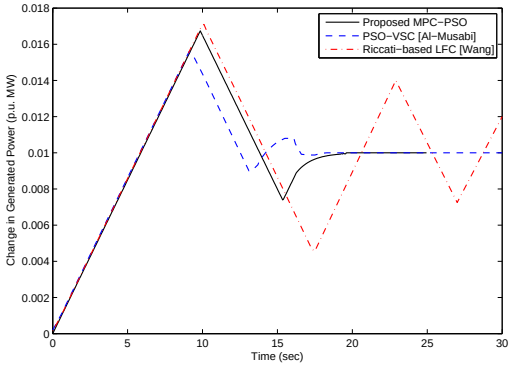

Figure 4.14: Generated Power Output for GRC = 0.0017

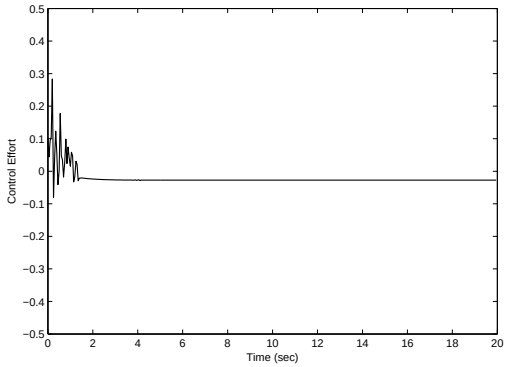

Figure 4.15: Control Effort for GRC = 0.01

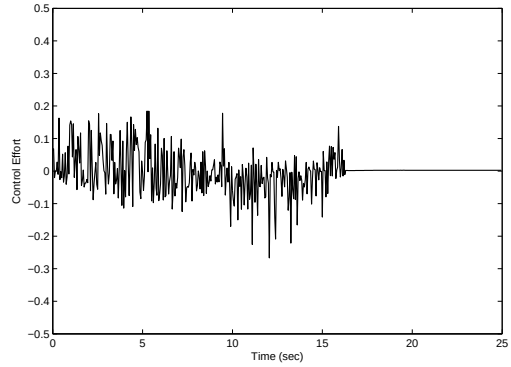

Figure 4.16: Control Effort for GRC = 0.0017

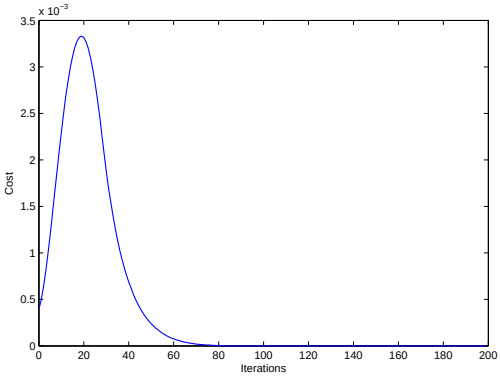

Figure 4.17: Cost for GRC = 0.01

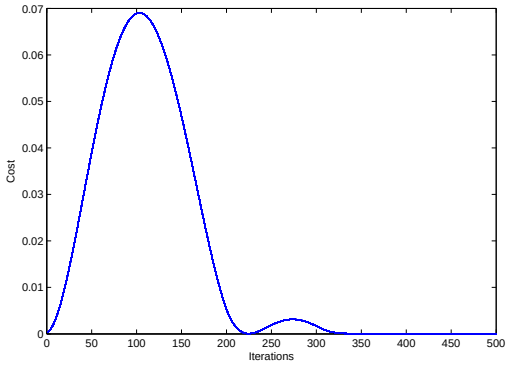

Figure 4.18: Cost for GRC = 0.0017

4.20 and 4.21 respectively. It is seen that the frequency deviates by 0.033 p.u. every time am incremental disturbance of 0.01 p.u. is given at the load. The frequency deviation is maximum at 0.07 p.u. when the load disturbance changes by 0.015 p.u. at 60s. The generated power from the system fulfils the load demand in all cases as seen from Figure 4.21. The control effort is seen in Figure 4.22, and it suggests that the maximum control inputs are applied when the load disturbance shifts to a new value. Slight fluctuations in the load frequency are adjusted by the control input.

The system is also tested for a range of GRC values by testing it for three cases. The values of GRC selected to be are 0.0017, 0.005 and 0.01. These GRC values are practical values and are dependent on the model and specifications of the power generation unit (turbine). All other parameters and control variables are same.

The results are seen in Figures 4.23 to 4.26. It is clear that the frequency deviation and the change in generated power, as well as the control effort applied is most for the case when the GRC is the smallest. The frequency deviates by as much as 0.152 Hz in this case and becomes 0 only after 19 seconds. The maximum value of the change in generated power is different in each case. It is 0.014, 0.016 and 0.017 for the

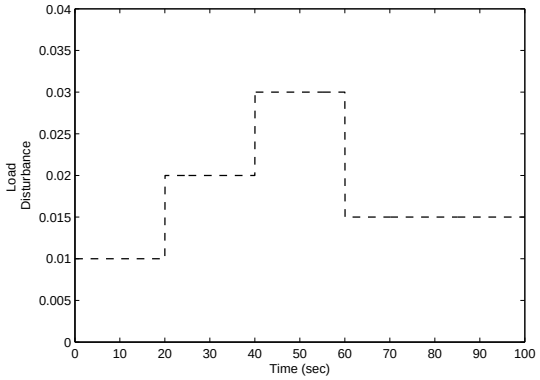

Figure 4.19: Varying Disturbance Applied

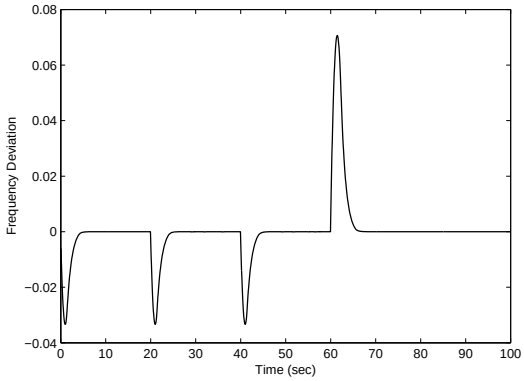

Figure 4.20: Frequency Deviation for Varying Disturbance

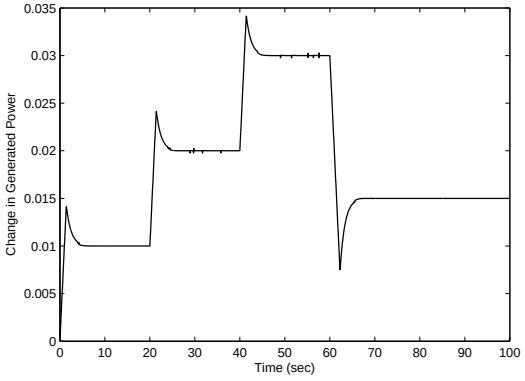

Figure 4.21: Generated Power Output for Varying Disturbance

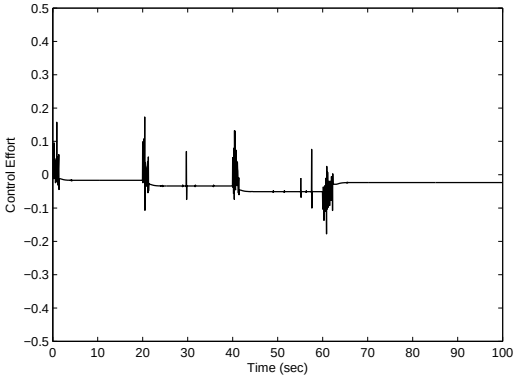

Figure 4.22: Control Effort for Varying Disturbance

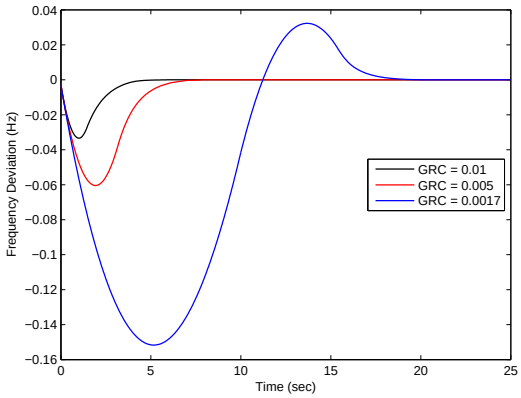

Figure 4.23: Frequency Deviation for Varying GRC

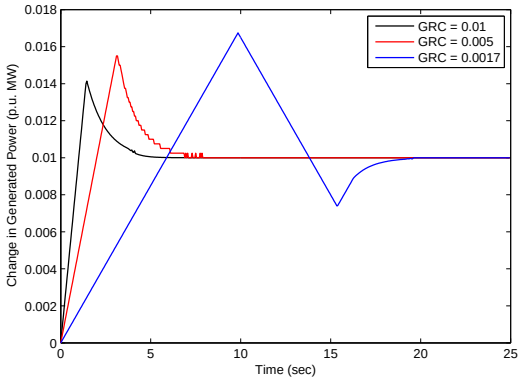

Figure 4.24: Generated Power Output for Varying GRC

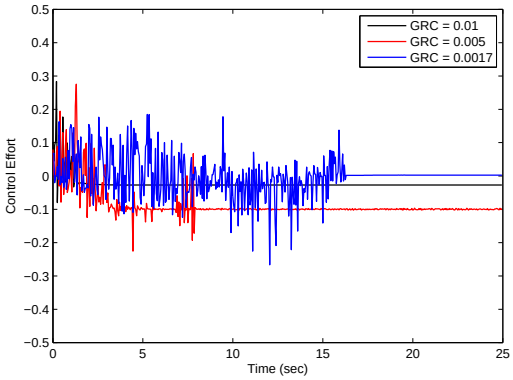

Figure 4.25: Control Effort for Varying GRC

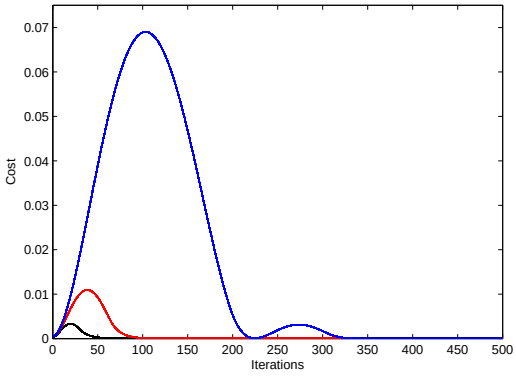

Figure 4.26: Cost Functions for Varying GRC

cases when GRC is 0.01, 0.005 and 0.0017 respectively. When the GRC is smallest at 0.0017 p.u., it takes longest, i.e. 20s for the system to provide the steady demand power of 0.01 p.u. MW. For the cases of GRC 0.01 and 0.005, it took 5 and 8 seconds respectively.

The control efforts seen in Figure 4.25, and they set at -0.0273, -0.1 and 0.0025 for the cases of GRC 0.01, 0.005 and 0.0017 respectively. The control inputs vary till the time it takes for the system to reach the required steady states, after which they take their steady states. The cost is also the most for the case of smallest GRC and least for the case with the largest.

4.3.3 LFC Including Nonlinearity and Parameter Variations

A challenging case involving two parts is considered here:

- 25% parameter variations in the system due to severe disturbances or modeling errors
- GRC nonlinearity of 0.01 p.u. $MW\ sec^{-1}$ applied on two states, x_2 and x_4

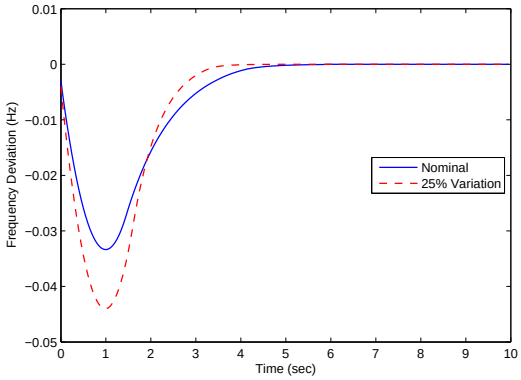

Figure 4.27: Frequency Deviation for GRC with Parameter Variation

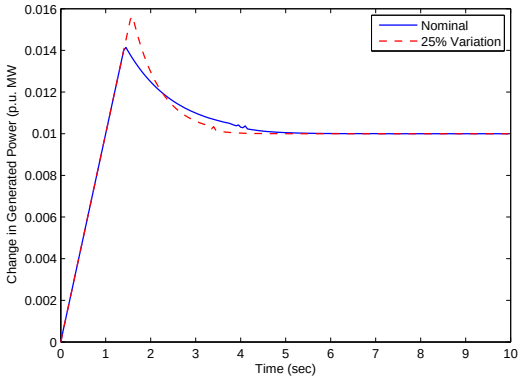

Figure 4.28: Generated Power Output for GRC with Parameter Variation

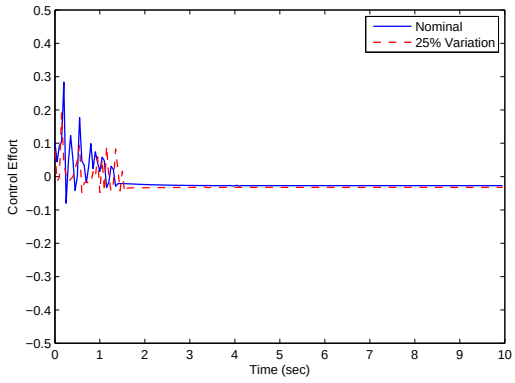

Figure 4.29: Control Effort for GRC with Parameter Variation

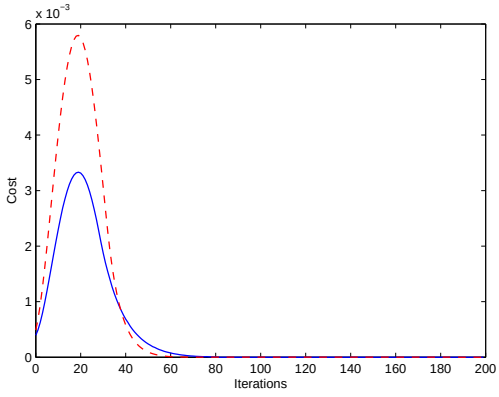

Figure 4.30: Control Effort for GRC with Parameter Variation

The corresponding values of A, B & F are:

$$
\begin{aligned}
A &= \begin{bmatrix} -0.0665 & 8 & 0 & 0 \\ 0 & -3.663 & 3.663 & 0 \\ -6.86 & 0 & -13.736 & -13.736 \\ 0.6 & 0 & 0 & 0 \end{bmatrix} \\
B &= \begin{bmatrix} 0 & 0 & 13.736 & 0 \end{bmatrix}^T \\
F &= \begin{bmatrix} -8 & 0 & 0 & 0 \end{bmatrix}^T
\end{aligned}
$$

The results of this comparison are given in Figures 4.32 to 4.35. It is seen that the frequency deviates by 33% more for the case when system parameters are varied by 25%. However, in this case it takes a little less time to reach the required value. The change in generated power almost remains the same for both cases. There is slight difference in the behavior which is clear from Figure 4.33. The change in generated power is observed to be 25% more for the case with parameter variation. It is also observed that the cost function is higher when the parameters are varying from the nominal values. The results indicate that the proposed controller is quite indifferent to the variation in system parameters.

4.4 Two-Area Load Frequency Control

In this section, the LFC problem is extended to two interconnected areas. The areas are connected as seen in Figure 4.31.

First the case of two-area LFC excluding nonlinearity is explored and then the GRC nonlinearity is also incorporated in the model to explore the nonlinear case.

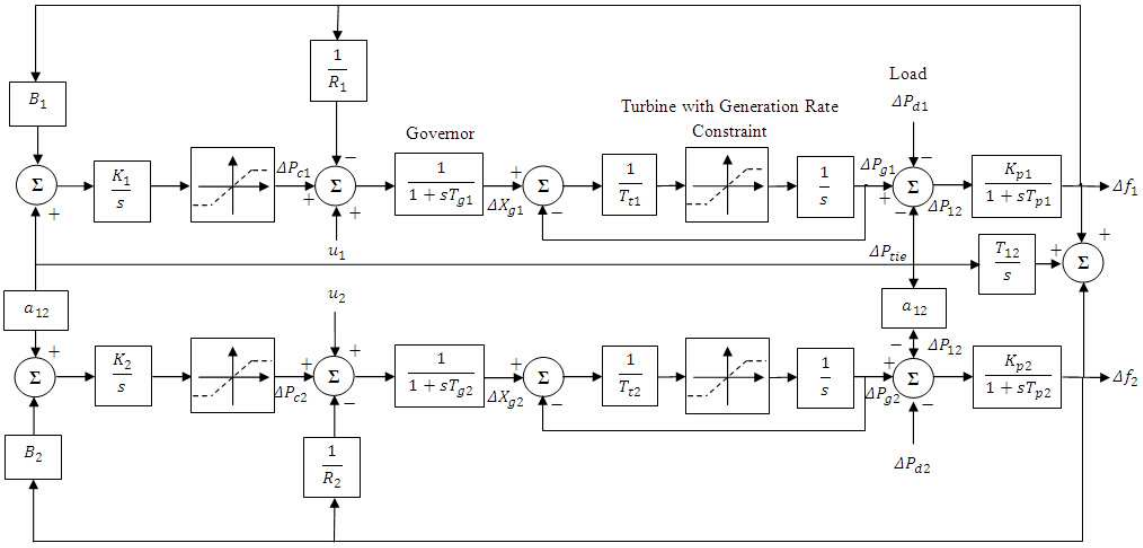

Figure 4.31: Block diagram of two-area LFC with GRC nonlinearities

The model in Figure 4.31 can be expressed by the following set of equations. The generalized model was given before also in Equation 4.1, but now it is extended for two-area system.

$$\dot{X}_i(t) = A_i x_i(t) + B_i u_i(t) + \sum_{j=1,\, j\neq i}^{n} E_{ij} x_j(t) + F_i d_i(t) \tag{4.10}$$

$$y_i(t) = C_i(t) x_i(t) \tag{4.11}$$

$$\dot{X}^T = [\Delta\dot{f}_1(t)\Delta\dot{P}_{g_1}(t)\Delta\dot{X}_{g_1}(t)\Delta\dot{P}_{c_1}(t)\Delta\dot{P}_{tie}(t)$$

$$\Delta\dot{f}_2(t)\Delta\dot{P}_{g_2}(t)\Delta\dot{X}_{g_2}(t)\Delta\dot{P}_{c_2}(t)] \tag{4.12}$$

$$A = \begin{bmatrix} \frac{-1}{T_{p1}} & \frac{K_{p1}}{T_{p1}} & 0 & 0 & \frac{-K_{p1}}{T_{p1}} & 0 & 0 & 0 & 0 \\ 0 & \frac{-1}{T_{t1}} & \frac{1}{T_{t1}} & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{-1}{R_1T_{g1}} & 0 & \frac{-1}{T_{g1}} & \frac{1}{T_{g1}} & 0 & 0 & 0 & 0 & 0 \\ -K_1B_1 & 0 & 0 & 0 & -K_1 & 0 & 0 & 0 & 0 \\ T_{12} & 0 & 0 & 0 & 0 & -T_{12} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{K_{p2}}{T_{p2}} & \frac{-1}{T_{p2}} & \frac{K_{p2}}{T_{p2}} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{-1}{T_{t2}} & \frac{1}{T_{t2}} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{-1}{R_2T_{g2}} & 0 & \frac{-1}{T_{g2}} & \frac{1}{T_{g2}} \\ 0 & 0 & 0 & 0 & K_2 & -K_2B_2 & 0 & 0 & 0 \end{bmatrix} \tag{4.13}$$

$$B^T = \begin{bmatrix} 0 & 0 & 1/T_{g1} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/T_{g2} & 0 \end{bmatrix} \tag{4.14}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{4.15}$$

The system is stable and the objective is to minimize the system frequency deviation $\Delta\dot{f}_1(t)$ and $\Delta\dot{f}_2(t)$ in Areas 1 and 2 respectively under load disturbances in both areas.

The parameters of the system are given below [166]:

$T_{p1} = T_{p2} = 20$s, $K_{p1} = K_{p2} = 120$ Hz p.u. MW^{-1}, $T_{t1} = T_{t2} = 0.3$s, $K_1 = K_2 =$

1 p.u. MW^{-1} rad^{-1}, $T_{g1} = T_{g2} = 0.08$s, $R_1 = R_2 = 2.4$ Hz p.u. MW^{-1} and $B_1 = B_2$ = 0.425 p.u. MW Hz^{-1}

Since the parameters in this model are identical, and the change in the tie-line power, ΔP_{tie} is caused by the difference in the area frequencies, $\Delta f_1(t)$ - $\Delta f_2(t)$, the performance of the system has been tested by applying the disturbance in Area 1 only.

4.4.1 Two-Area LFC Excluding Nonlinearity

In this case, the GRC nonlinearity is excluded. A step disturbance of 0.03 p.u. is applied constantly on the system in Area 1. The cost function in this case is taken to be as follows:

$$J = \sum_{i=1}^{H_p} \Delta f_1(t)^2 + \Delta f_2(t)^2 + \Delta P_{tie}^2 + \Delta P_{g_1}^2 + \Delta P_{g_2}^2$$

Such a cost function ensures that the system is internally stable. The terms of the cost function are scaled equally. The control signals in this case are constrained stringently to be -0.1 $\leq$ u $\leq$ 0.1. Since the control signal is already so much constrained within its maximum limit, there is no limit on the change of control, Δu.

The dynamics of the system in this case are given in Figures 4.32 to 4.35 and the results are compared with previous work [4] as well as LFC using the pole placement technique. The behavior of the frequency deviation in both areas is seen as well as the change in generated power in both areas. In comparison with the PSO-VSC technique two cost functions are compared. J_1 is the same cost function used here, while J_2 proposed in [4] is a slightly different cost function as it incorporates the

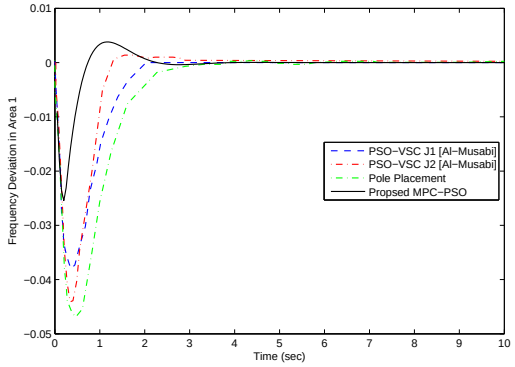

Figure 4.32: Frequency Deviation in Area 1

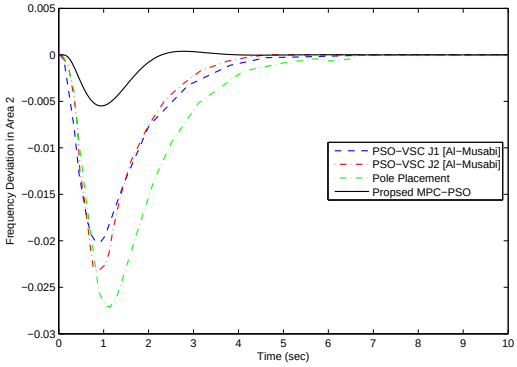

Figure 4.33: Frequency Deviation in Area 2

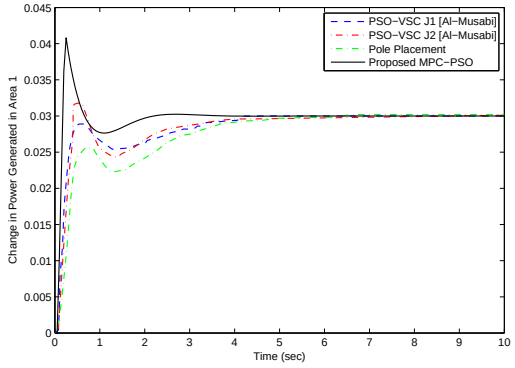

Figure 4.34: Change in Generated Power in Area 1

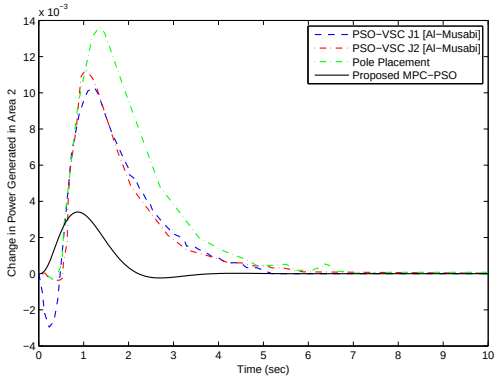

Figure 4.35: Change in Generated Power in Area 2

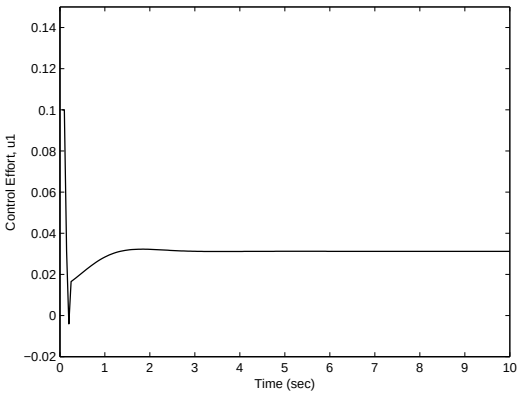

Figure 4.36: Control Effort, u_1

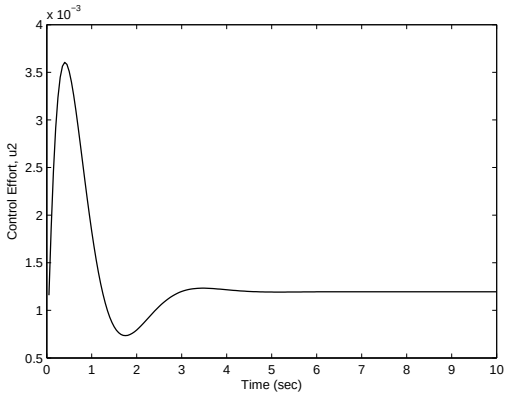

Figure 4.37: Control Effort, u_2

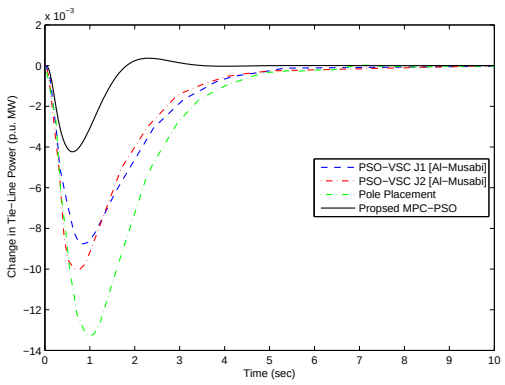

Figure 4.38: Power Flow in Tie-Line

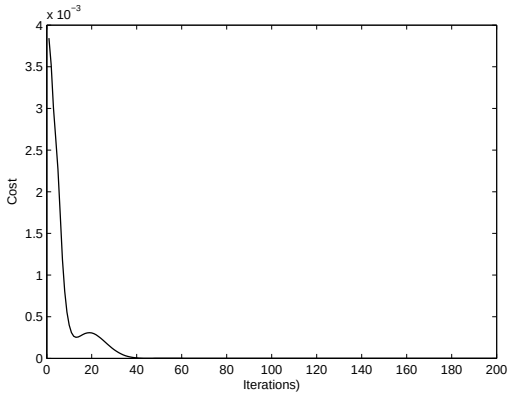

Figure 4.39: Cost Function

control inputs into it as well. The results show the the proposed technique performs much better in all aspects as compared with the previous work. It is seen in Figure 4.32 that the frequency deviation in Area 1 is less than what it was in the previous work as well as using the pole placement technique. Also, the frequency deviation becomes zero quicker using the proposed technique, than using previous techniques. From Figure 4.33, it is seen that for the Area 2, the deviation is at least 75% lesser compared to previous work.

It is seen in Figure 4.34 that the system is supplying the required 0.03 p.u. load from Area 1. The required load is supplied much quickly than in the compared techniques. But the trade-off for it is that the maximum change in generated power using the proposed technique is 0.01 p.u. more than previous work. There is minor deviation of generated power in Area 2, and it is must less compared to the deviation that is observed using other techniques. After that, the change in generated power in Area 2 becomes zero. Figure 4.38 shows the change in the tie-line power flow. Due to the frequency deviation in both areas, power begins to flow in the tie-line and as soon as the frequency deviations reach 0, the power flow in the tie-line also stops.

Compared to previous work and the pole placement technique, the power flow in the tie-line is much less using the proposed technique. The maximum flow in the tie-line in this case is -0.004 p.u., while for the pole placement technique, it is about 0.013 p.u.

Figures 4.36 and 4.37 show the control efforts of the system. It is observed that the control effort obeys the constraints and the controller provides the optimal control inputs to the system to enable it to cope with the continuous disturbance of 0.03 p.u. The cost function is given in Figure 4.39.

Now a more challenging case of two-area LFC excluding nonlinearity is studied. In this case the parameters of the two areas differ in the following respects:

$T_{p1} = 25$s, $T_{p2} = 20$s, $K_{p1} = 112.5$ Hz p.u. MW^{-1}, $K_{p2} = 120$ Hz p.u. MW^{-1}.

The rest of the system parameters are identical. This two-area system is subjected to a huge disturbance of 0.1 p.u. in both areas. This case is taken from the paper by Kong [83], in which State Contractive Constraint (SCC)-based MPC is applied to the LFC problem. The comparison of the proposed technique with SCC-MPC is given in Figures 4.40 to 4.43. It is seen that the proposed MPC-PSO technique gives a much smoother control of the system. From Figure 4.40 it is clear that although the SCC-MPC is able to bring the Area 1 frequency deviation to zero 1 second earlier than MPC-PSO, there are a lot of oscillations and there is also a steady state error using the SCC-MPC technique. The proposed technique enables the Area 1 frequency deviation to become zero more smoothly and accurately. The frequency deviation in Area 2 is seen in Figure 4.41. It is seen that proposed technique fares enormously better than

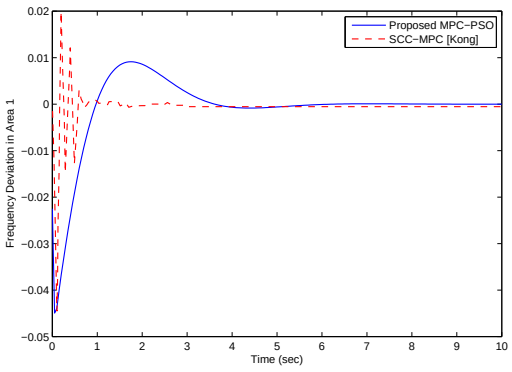

Figure 4.40: Frequency Deviation in Area 1

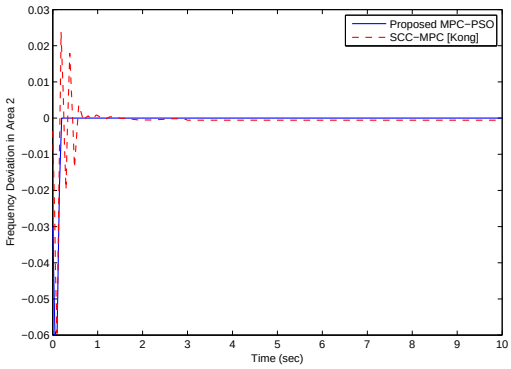

Figure 4.41: Frequency Deviation in Area 2

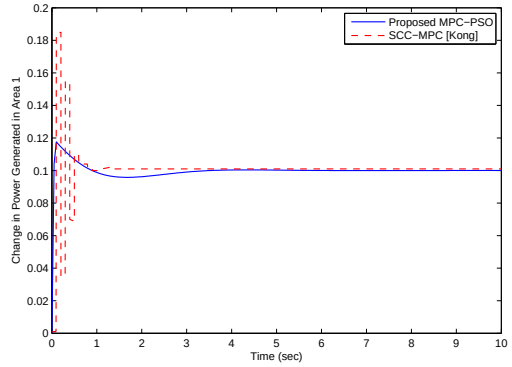

Figure 4.42: Change in Generated Power in Area 1

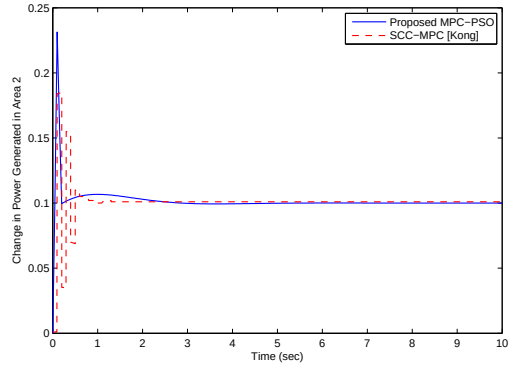

Figure 4.43: Change in Generated Power in Area 2

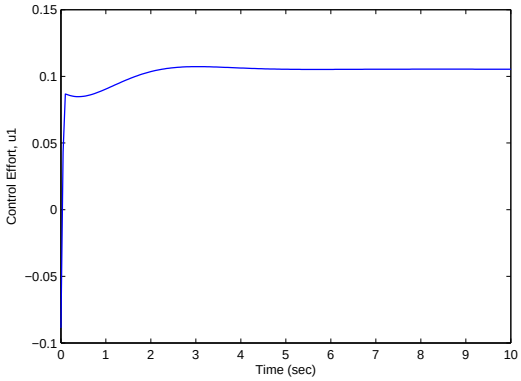

Figure 4.44: Control Effort, u_1

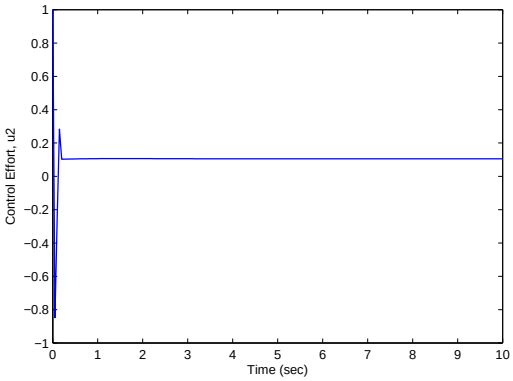

Figure 4.45: Control Effort, u_2

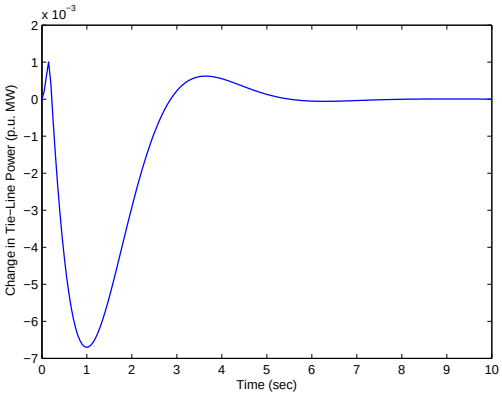

Figure 4.46: Power Flow in Tie-Line

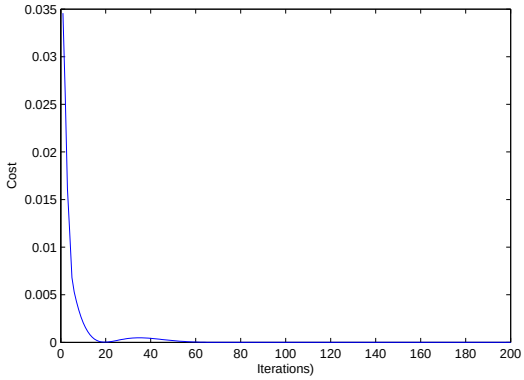

Figure 4.47: Cost Function

SCC-MPC. The change in generated power from the areas is seen in Figures 4.42 and 4.43. It is seen that the proposed techniques enables the system to cope with the power demand more smoothly, with lesser overshoot and shorter duration without any steady state errors. Since [83] does not give any details on the control constraints, they are taken to be -0.5 $\leq$ u $\leq$ 0.5 and the result is shown in Figures 4.44 and 4.45. It is seen that optimal control efforts needed to continuously cope up with the power demand are generated by the controller and applied to the system. The power flow in the tie-line is shown in Figure 4.46. It is seen that after 5s, once all the required states of the system are at equilibrium, the power in the tie-line also becomes zero.

4.4.2 Two-Area LFC Including GRC Nonlinearity

In this case, a GRC nonlinearity of 0.015 p.u. is incorporated in the system [4], [166]. The model of the system is given in Figure 4.31. The cost function used is as follows:

$$J = \sum_{i=1}^{H_p} \Delta f_1(t)^2 + \Delta f_2(t)^2 + \Delta P_{tie}^2 + \Delta P_{g_1}^2 + \Delta P_{g_2}^2$$

The system is subjected to a 0.01 p.u. disturbance. The results for this case are given in Figures 4.48 to 4.54. It is seen that the system is brought to equilibrium in about 5.5s and the power flow in the tie-line between the areas also becomes zero after that time, as seen in Figure 4.54. Figure 4.48 compares the performance of the proposed controller to several techniques available in the literature for this problem. The comparisons are given with LQR, Optimal Integral Controller, PSO-VSC [4], and Decentralized VSC technique for LFC proposed by Yang [166]. The comparison with the LQR is significant as it is the benchmark controller. However, it utilizes all system states while the proposed technique utilizes the practically available output states only [43].

It is seen that the proposed technique is closely comparable with the LQR controller and out-performs the techniques given in previous work. As a result, the freuquency deviation in Area 1 becomes zero in less than 5 seconds with lesser oscillations as seen in Figure 4.48. For the Area 2, it is seen from Figure 4.49 that the frequency deviation is much less compared to PSO-VSC [4] and Optimal Integral Controller cases. Thus the proposed controller is able to minimize the frequency deviation in an adjacent area better. The system is able to supply the load with the required 0.01 p.u. increase in power demand from Area 1, while the power generated from Area 2 remains unchanged. The control efforts shown give the optimal control input that is applied continuously to keep the system at steady state while coping with the load disturbance. Comparing the power flow in tie-line in Figure 4.54, it is seen that the power deviation in it becomes zero quicker than in other techniques and closely

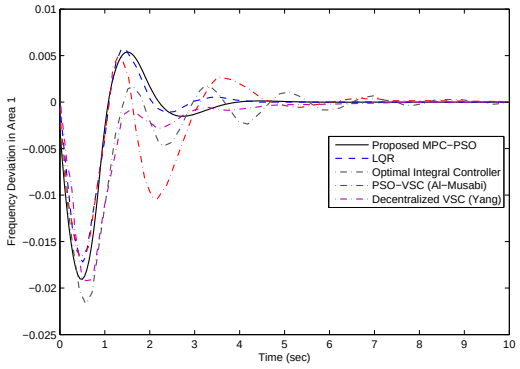

Figure 4.48: Frequency Deviation in Area 1

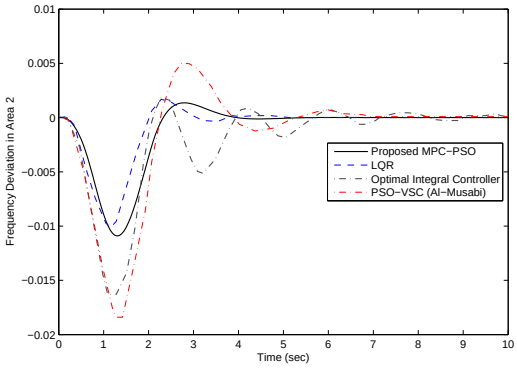

Figure 4.49: Frequency Deviation in Area 2

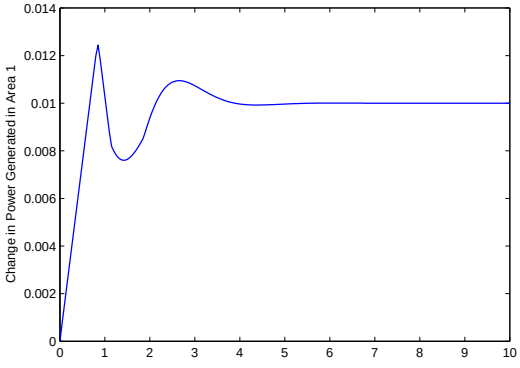

Figure 4.50: Change in Generated Power in Area 1

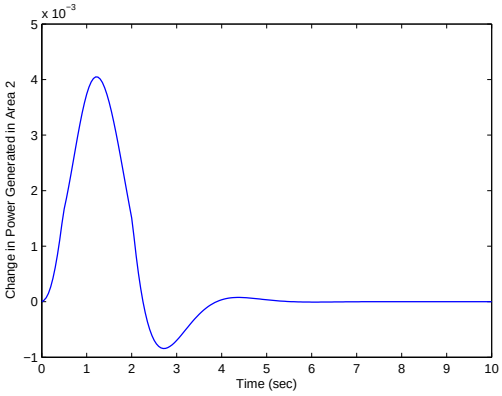

Figure 4.51: Change in Generated Power in Area 2

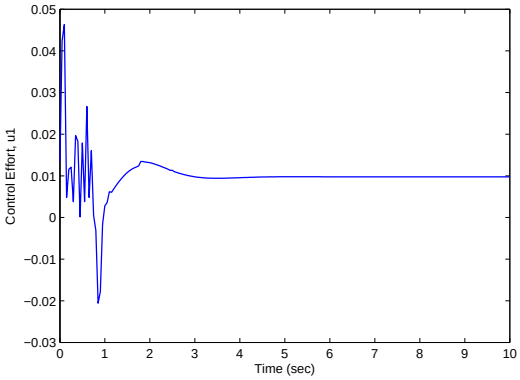

Figure 4.52: Control Effort, u_1

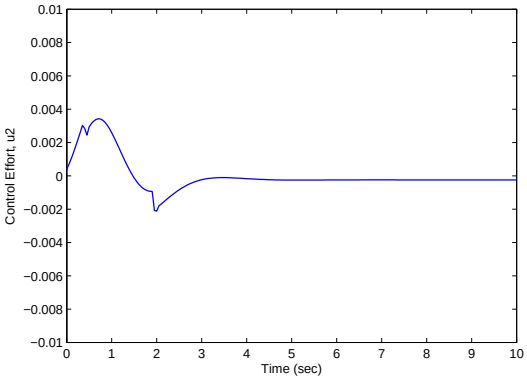

Figure 4.53: Control Effort, u_2

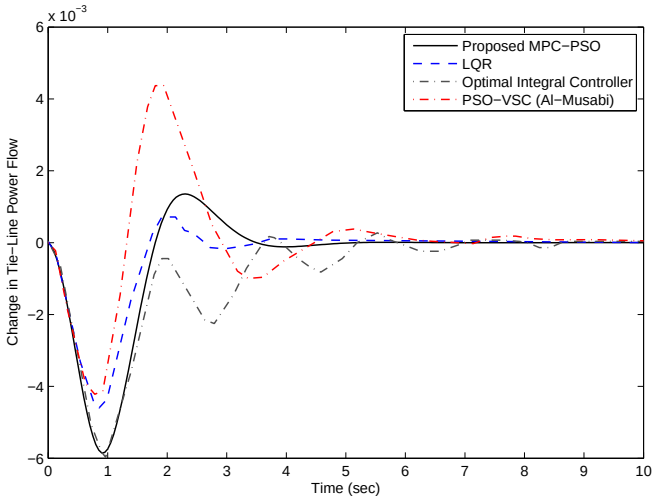

Figure 4.54: Power Flow in Tie-Line

compares with LQR.

4.5 Four-Area Load Frequency Control

A multiarea interconnected system is considered now. It is taken to be connected as a combination of ring as well as longitudinal manners [166], as shown in Figure 4.55.

This can be more clearly seen in the block diagram of the first area of this system in Figure 4.56. The other areas of the system can also be represented by similar block diagrams.

The nominal parameter values for this system are:

$T_{p1} = 20\text{s}$, $T_{p2} = 25\text{s}$, $T_{p3} = 20\text{s}$, $T_{p4} = 15\text{s}$

$K_{p1} = 120$, $K_{p2} = 112.5$, $K_{p3} = 125$, $K_{p4} = 115$ Hz p.u. MW^{-1}

$T_{t1} = 0.3\text{s}$, $T_{t2} = 0.33\text{s}$, $T_{t3} = 0.35\text{s}$, $T_{t4} = 0.375\text{s}$

$K_1 = K_2 = K_3 = K_4 = 0.6$ p.u. $MW^{-1}\ rad^{-1}$, $T_{g1} = 0.08\text{s}$, $T_{g2} = 0.072\text{s}$, $T_{g3} =$

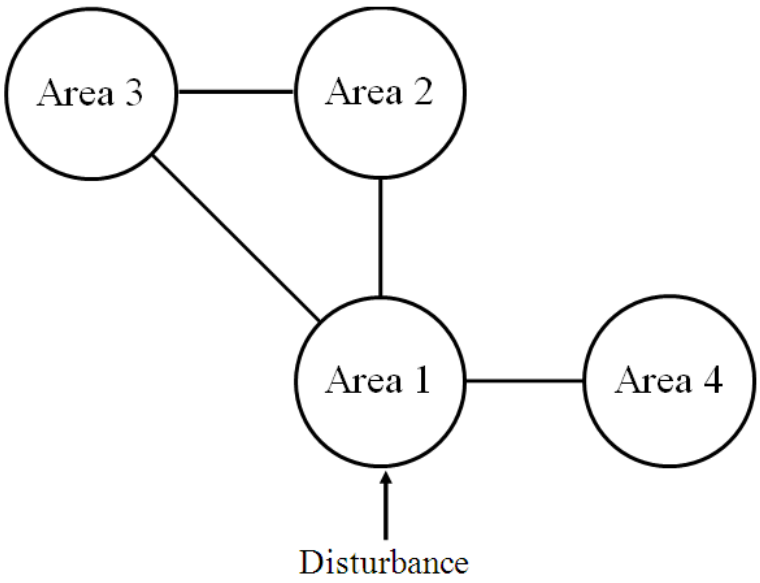

Figure 4.55: Block diagram of four-area interconnected power system

0.070s, $T_{g4} = 0.085$s

$R_1 = 2.4$, $R_2 = 2.7$, $R_3 = 2.5$, $R_4 = 2$ Hz p.u. MW^{-1}

$T_{12} = T_{13} = T_{14} = T_{21} = T_{23} = T_{31} = T_{32} = T_{41} = 0.545$

$T_{24} = T_{34} = T_{42} = T_{43} = 0$

$B_1 = B_2 = B_3 = B_4 = 0.425$ p.u. MW Hz^{-1}

The model of the system can be generalized as given previously in Section 4.2.

In this case as well, a step load disturbance of 0.03 p.u. is applied to Area 1 of the system for the case when nonlinearity is excluded and a load disturbance of 0.01 p.u. is applied for the case including the GRC nonlinearity. The cost function used in this case is a condensed form of the previously used one, to enable emphasis on the most important states of the system only. Thus it is:

$$J = \sum_{i=1}^{H_p} \Delta f_i(t)^2 + \Delta P_{g_1}^2$$

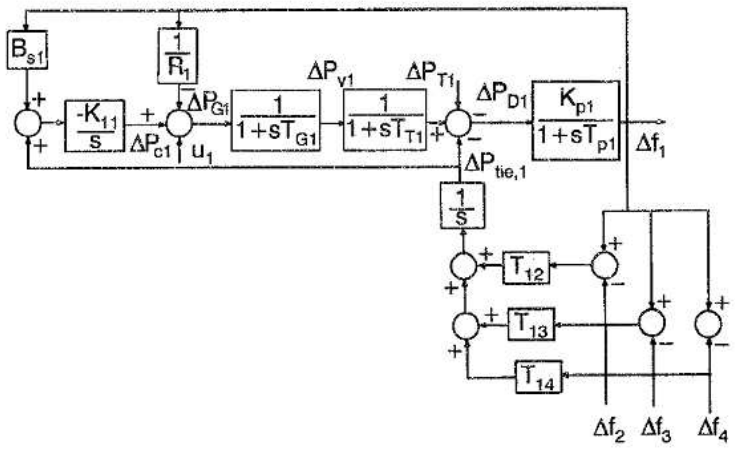

Figure 4.56: Area 1 of four-area interconnected power system

For the case when the nonlinearity is excluded, the results are given in Figures 4.57 to 4.63. It is seen that the frequency deviation in all areas is controlled to be zero and the power demand is met for Area 1. The power generated for all other areas remains unchanged, although they experience minor fluctuations while the power demand in Area 1 is being met. The power fluctuates in the tie-lines also due to the step load disturbance, but eventually become zero as the system reaches steady states. The power flow in the tie-lines can be seen in Figure 4.63. The optimal control efforts are seen in Figures 4.61 and 4.62.

For the case including nonlinearity, a GRC value of 0.1 p.u. MW/s is applied to the system and it is subjected to a step load disturbance of 0.01 p.u. The results for this test are given in Figures 4.64 to 4.70. A comparison of these results with the linear case, with 0.01 p.u. disturbance is also given in these figures. The effect of nonlinearity is clearly seen. The frequency deviation and change in generated power is obviously more in the case with nonlinearity. Also, the control efforts applied to

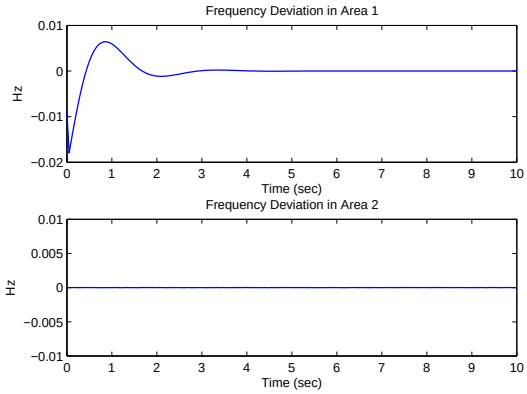

Figure 4.57: Four-Area Excluding Nonlinearity, Δf_1 & Δf_2

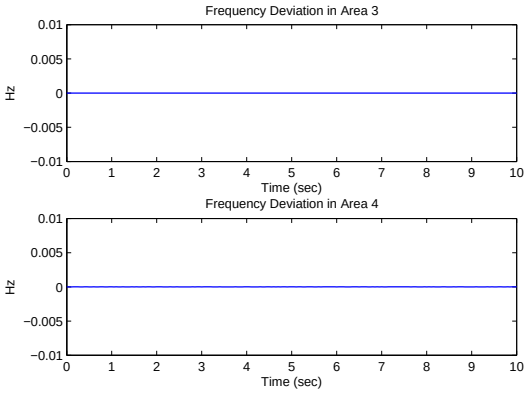

Figure 4.58: Δf_3 & Δf_4

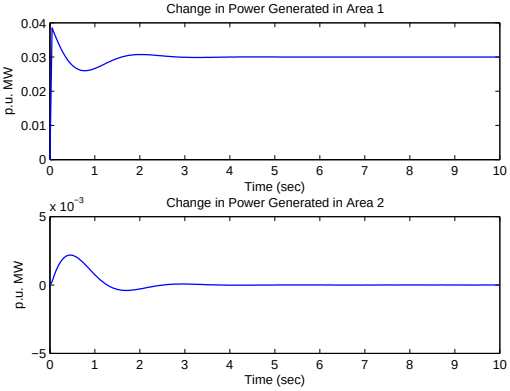

Figure 4.59: ΔP_{g1} & ΔP_{g2}

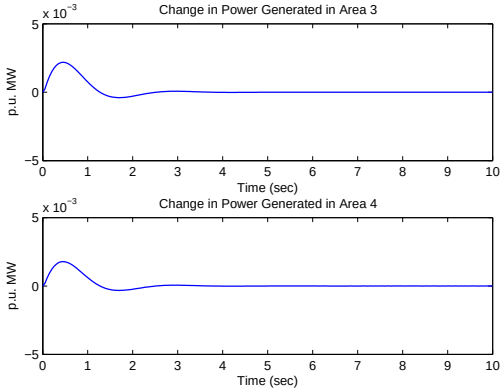

Figure 4.60: ΔP_{g3} & ΔP_{g4}

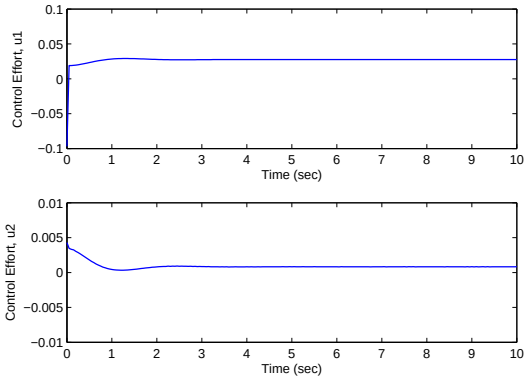

Figure 4.61: u_1 & u_2

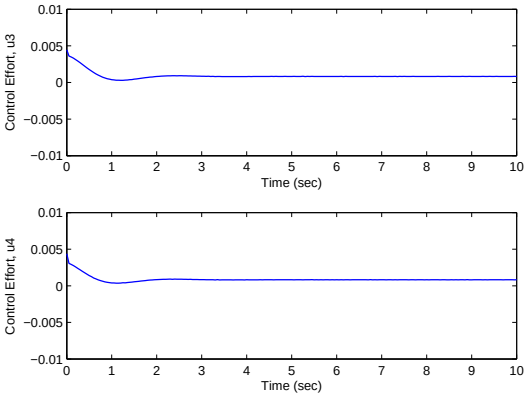

Figure 4.62: u_3 & u_4

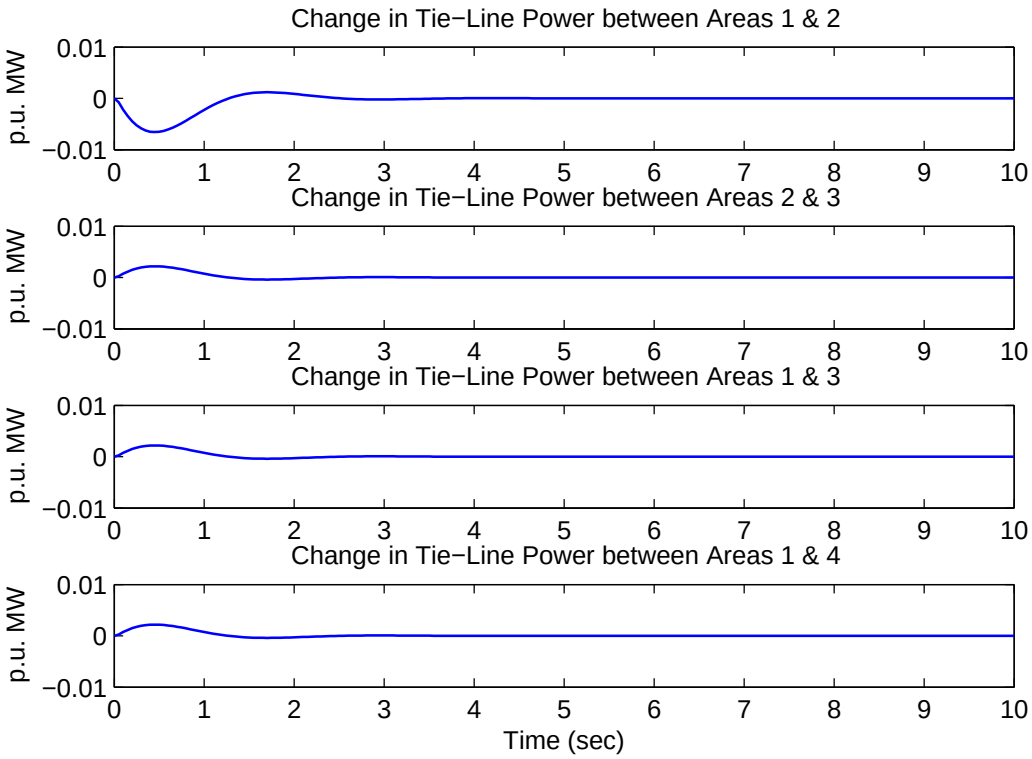

Figure 4.63: Power Flow in Tie-Lines for Four-Area System Excluding GRC

the system are slightly greater in this case.

4.6 Conclusion

The proposed controller is applied to the challenging LFC problem. Both linear and nonlinear cases are explored and the results show that the controller is able to successfully control the load-frequency of single as well as interconnected power systems. The proposed controller works well for several linear and nonlinear cases, with varying values of the nonlinearity as well as parameter variations. Both SISO (single area) and MIMO (two and four area) systems are able to be controlled using this technique. Comparison with published literature show improvements in the load-frequency control by using this newly proposed MPC-PSO controller for power systems. Several examples from the existing literature are compared with the performance of the proposed controller on this problem for both linear and nonlinear cases, as well as for single and two-area LFC systems. The consistent quality of the results over several different cases suggest that the proposed method is robust and reliable for the LFC problem. Since a general methodology of controller design is presented, controllers for any n-area LFC system can be developed using this technique.

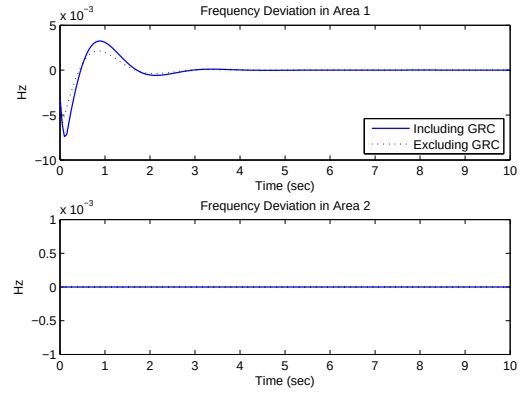

Figure 4.64: Four-Area Including Nonlinearity, Δf_1 & Δf_2

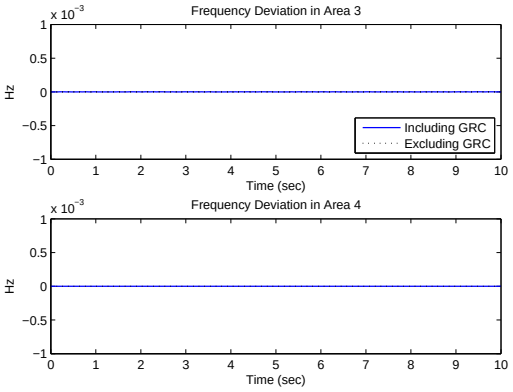

Figure 4.65: Δf_3 & Δf_4

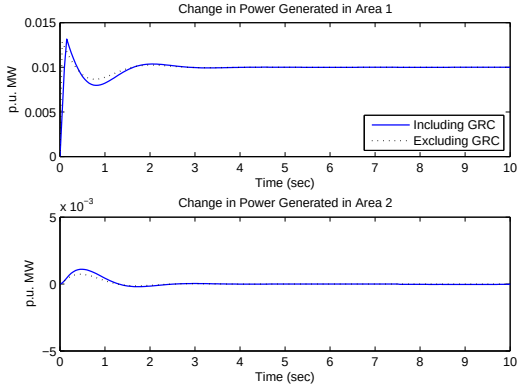

Figure 4.66: ΔP_{g1} & ΔP_{g2}

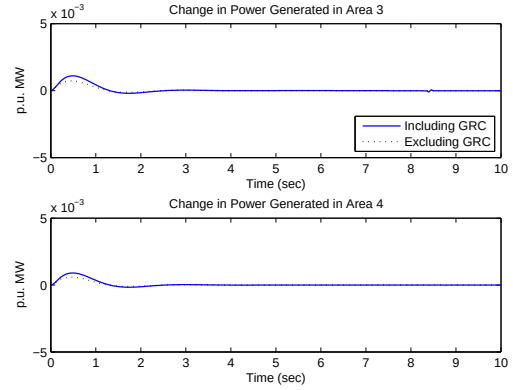

Figure 4.67: ΔP_{g3} & ΔP_{g4}

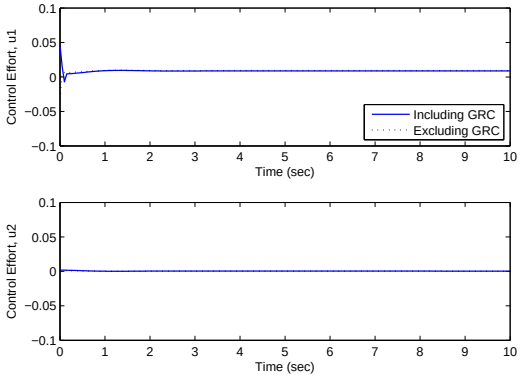

Figure 4.68: u_1 & u_2

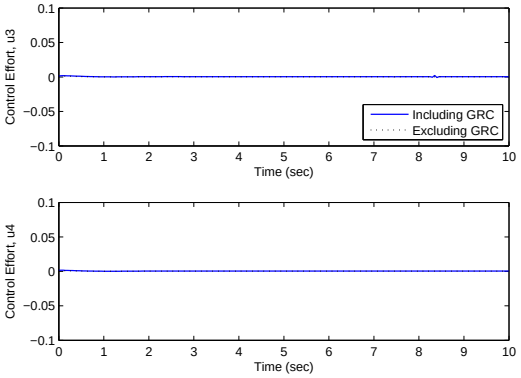

Figure 4.69: u_3 & u_4

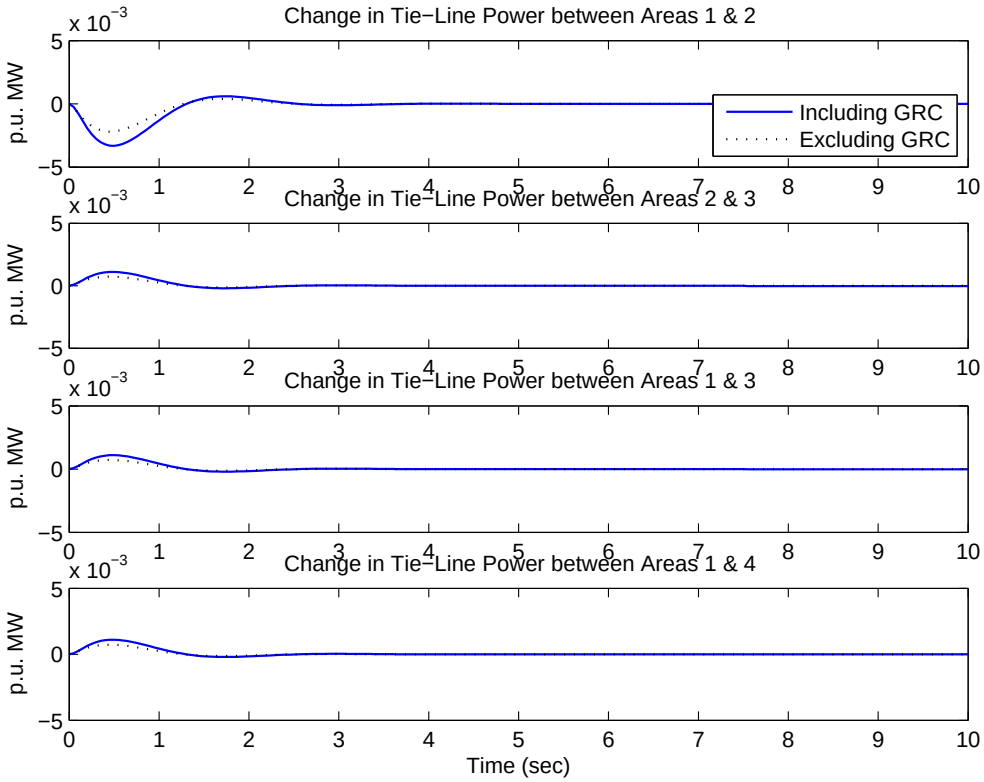

Figure 4.70: Power Flow in Tie-Lines for Four-Area System with GRC

CHAPTER 5

APPLICATION OF MPC-PSO TO A FOSSIL FUEL POWER UNIT

5.1 Introduction

Power generation units can be fueled by several means, but one of the most common means is using fossil fuels, typically diesel or natural gas. A Fossil Fuel Power Unit (FFPU) essentially consists of a boiler and a turbine. The boiler is used to generate steam which drives the turbine, these generating electrical power. The steam is produced by heating water in a furnace. The temperature of the furnace is controlled by controlling the flow of the fossil fuel.

The boiler-turbine operation is complex as steam must be delivered at a constant rate to the coupled turbine for reliable operation. The rate of steam delivered has to be altered to adjust the speed of the turbine for changing the generated power capacity. At the same time, it is also desired to keep the boiler pressure constant

and water level deviation minimum. Thus it manifests itself as a highly nonlinear and strongly coupled complex problem with uncertainties. It is critical to control this process efficiently to optimize the use of fuel, minimize losses and maximize the profitability of operation through the selection of optimal control signals.

The fossil fuel boiler-turbine power generation system has been studied by many researchers. However, there is a lack of published literature in this area as a bulk of the work on optimizing boiler-turbine operation and maximizing fuel efficiency is probably proprietary or patented. However we do find some discussion on the control of this system.

There are several types of boiler-turbine models in the literature with varied complexity levels and orders, depending upon the number of factors considered during modeling. A generalized drum-boiler-turbine model is considered here. This was originally developed by Åström and Eklund [1] by studying the power generation plant P16/G16 at Sydvenska Kraft AB plant at Malmö, Sweden. The model was published in 1972 using data collected in 1969. The model was presented as a second order nonlinear system. Morton and Price updated this model in 1977 by incorporating water level deviations [99], however the results were not good [42]. Finally, this model was updated by Bell and Åström in 1987 and this is the one used here. This model was also used in [31].

A classical adaptive control scheme is applied to this model by Hogg [67] using multiloop PI controller. The boiler-turbine parameters were considered to be time-varying. This was further extended by Tu by comparing different multivariable

schemes for this problem [148]. Chen and Shamma [31] used linear programming for online control of this system. The control signals were generated by formulating the problem as l^1-optimal control problem. It is found that it is a challenge to design a controller for this system that can work for all operating points. This problem was also addressed by Dimeo and Lee using Genetic Algorithm to design a universal coupled PI controller that can work at all the operating points of this system [42]. In 2004, Tan *et. al.* [145] applied a linear controller to this problem using a distance measure via the gap metric technique. The gap metric technique is used to compute the distance between the nonlinear boiler-turbine system and a fixed linear system. To tackle the strong nonlinearities of this system, they had to use only certain operating points, otherwise the system would not have worked well. Moreover, the could not keep the water level deviation zero, as required by the system at any operating point. Clearly, this method well worked for some selected operating points, but the actual operating points given by Bell and Åström [13] were not entirely controllable.

A slightly different model of this FFPU system is given by Garduno-Ramirez [51]. In his thesis, he has applied several intelligent control techniques to this system, however, the response of this model appears to be much slower compared to Åström's model. Work by Heo [65] and Chang [28] use this model to applied Multiobjective and Multiagent control techniques respectively. However, the response is very slow and is studied for near operating points only. Xiao and Wang [161] applied Neural Network based MPC to this model, but mainly considered very close switching points themselves and response for switching to far operating points is extremely slow, taking

up to 20 mins. Shakil [128] applied a Tabu-Search based predictive controller to this system that enable good control through the complete operating range. The results showed smoother switching between operating points and low water level deviations while the control signals were also smoothly changing.

The proposed controller will be applied to the excellent and well described model given by Bell and Åström and will tackle the nonlinear system directly. Moreover, the full range of operating points will be studied. A great advantage of using nonlinear control is that any intermediate operating point can be also given to the system. Therefore, it is not necessary to switch to the required operating points only. However, as this is the method by which this system is operated, the system will be operated at these defined operating points.

5.2 Model of Drum-Boiler-Turbine Power Unit

There are many types of boiler-turbine models in the literature with varied complexity levels and orders, depending upon the number of factors considered during modeling. A generalized drum-boiler-turbine model is considered here. This was originally developed by Åström and Eklund [1] by studying the power generation plant P16/G16 at Sydvenska Kraft AB plant at Malmö, Sweden. The model was published in 1972 using data collected in 1969. The model was presented as a second order nonlinear system. Morton and Price updated this model in 1977 by incorporating water level deviations [99], however the results were not good [42]. Finally, this model was updated by Bell and Åström in 1987 and this is the one used here. This model was also used in [31].

The fossil fuel boiler-turbine system is modeled as a third order nonlinear MIMO system, represented by the following differential equations.

$$\dot{y}_1 = -0.0018u_2y_1^{9/8} + 0.9u_1 - 0.15u_3 \tag{5.1}$$

$$\dot{y}_2 = (0.073u_2 - 0.016)y_1^{9/8} - 0.1y_2 \tag{5.2}$$

$$\dot{y}_4 = \frac{(141u_3 - (1.1u_2 - 0.19)y_1)}{85} \tag{5.3}$$

$$L = y_3 = 0.05(0.13073y_4 + 100\alpha_{cs} + \frac{q_e}{9} - 67.975) \tag{5.4}$$

Where, L is the water level and α_{cs} and q_e are evaporation rate (Kg/s) and steam quality variables respectively, which are given by

$$\alpha_{cs} = \frac{(1 - 0.001538y_4)(0.9y_1 - 25.6)}{y_4(1.0394 - 0.0012304y_1)} \tag{5.5}$$

$$q_e = (0.854u_2 - 0.147)y_1 + 45.59u_1 - 2.51u_3 - 2.096 \tag{5.6}$$

This system can be represented by a block diagram as in Figure 5.1.

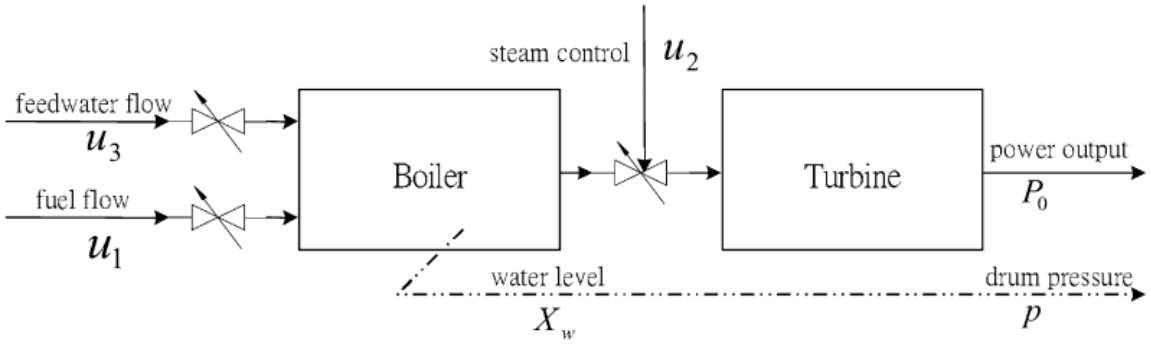

Figure 5.1: Block diagram of Drum Boiler Turbine System

The control signals, u_1, u_2 and u_3 are given to the three control valves. The are

described below:

- u_1 is the fuel flow to the combustion unit of the boiler. Although the combustion process is very complex and depends on several factors, the most important factor is the fuel flow into the combustion chamber or furnace. The system is also modeled based on that.

- u_2 is the steam flow to the turbine coupled to the boiler. The amount of steam is responsible for the speed of the turbine.

- u_3 controls the flow of feedwater into the furnace.

The states of the system are described as:

- y_1 is the steam pressure, P in Kg/cm^2.

- y_2 is the electrical power generated, P_o from the turbine in MW.

- y_3 is the water level deviation, X_w in m.

- y_4 is the fluid density in Kg/m^3.

Inputs to the boilers given in the model are normalized between [0,1]. The rate limitations on the inputs are given below:

$$\begin{aligned} |\dot{u_1}| &\leq 0.007/\text{sec} \\ -2/\text{sec} \leq |\dot{u_2}| &\leq 0.02/\text{sec} \\ |\dot{u_3}| &\leq 0.05/\text{sec} \end{aligned} \tag{5.7}$$

5.3 Operating Points of Drum-Boiler-Turbine Power Unit

The system must be operated at the specified operating points. Table 6.2 shows the different operating points of the boiler. Initially, the boiler has to be started at the nominal operating point (OP# 4), i.e. at 100% and this has to be done manually (open loop). After the boiler reaches the nominal operating point, the controller is switched on and the system becomes closed loop. The operation of the boiler can be switched between these operating points after that, and also to an intermediate operating point. It is not practical to switch from the nominal operating point to an extreme operating point, i.e. from OP# 4 to OP# 1 and OP# 7 directly. Such a demand might lead to unstability.

Table 5.1: Operating Points for the System

OP#:	1(70%)	2(80%)	3(90%)	4(nom,100%)	5(110%)	6(110%)	7(130%)
y_1	75.6	86.4	97.2	**108**	119	130	140
y_2	15.3	36.7	50.5	**66.65**	85.1	105	128
y_4	300	342	385	**428**	472	513	556
u_1	0.15	0.20	0.27	**0.34**	0.41	0.50	0.60
u_2	0.48	0.55	0.62	**0.69**	0.75	0.82	0.89
u_3	0.18	0.25	0.34	**0.43**	0.54	0.66	0.79

5.4 Problem Formulation

The objective is to design an MPC-PSO controller for the boiler-turbine system that can provide stable and efficient switching between different operating points. The goal is to switch the operating points:

- In minimum time

- With minimum steady state error
- With minimum overshoots and oscillations
- With minimum water level deviation

The MPC-PSO controller is applied directly to the model. It is assumed that the model is known, as given in [13]. A sampling time of 1 sec is used for the system. Since all control signals can affect all the system states, a coupled scheme is used. A prediction horizon, H_p of 15 is used for each control input. PSO particles are taken to be 25, and the other PSO parameters are the same as taken in previous examples. A time varying weight is used.

5.5 Operation at Nominal Operating Points

Figures 5.2 to 5.7 show the operation of the system at nominal operating points. The control is applied directly to the nonlinear model of the system and all the constraints on the control are also implemented. The MPC-PSO controller successfully generates the optimal signals required to keep the boiler operating constantly at this operating point. Comparing with the data in Table 6.2, the values of the control efforts, u_1, u_2, and u_3 are maintained 0.34, 0.69 and 0.43 respectively. It is observed that the power, pressure and water level are constant also.

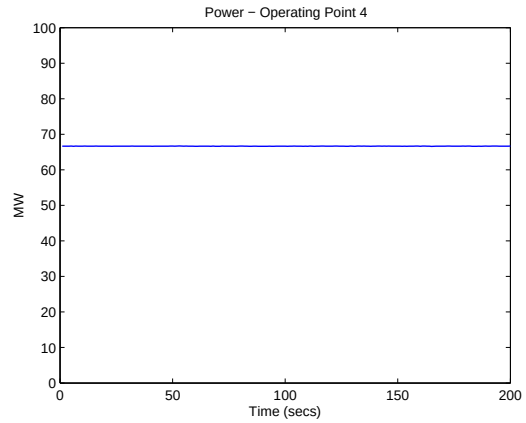

Figure 5.2: Power Generated at Operating Point 4

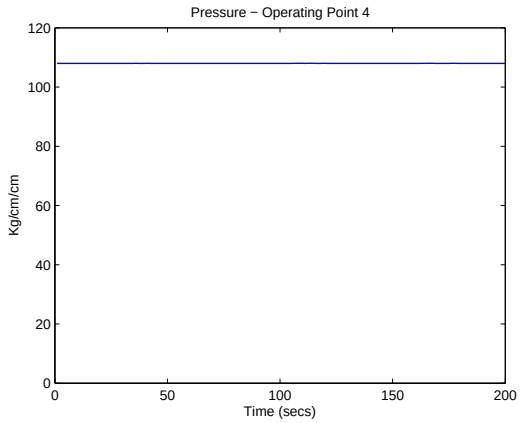

Figure 5.3: Pressure at Operating Point 4

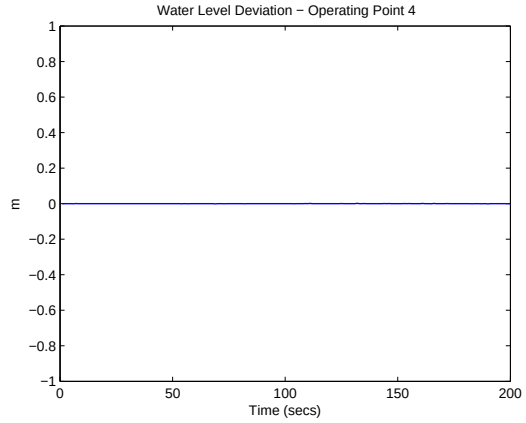

Figure 5.4: Water Level at Operating Point 4

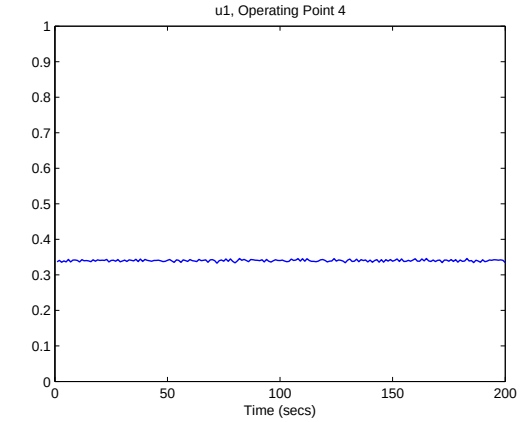

Figure 5.5: u_1 at Operating Point 4

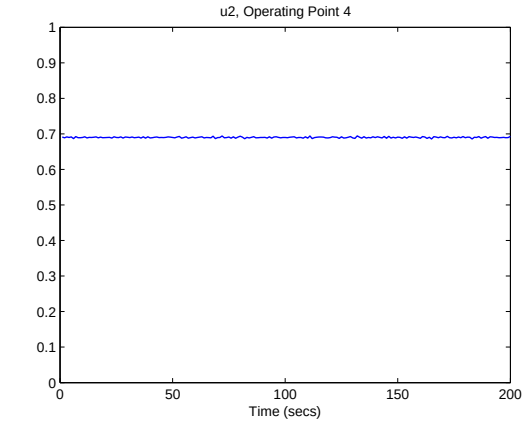

Figure 5.6: u_2 at Operating Point 4

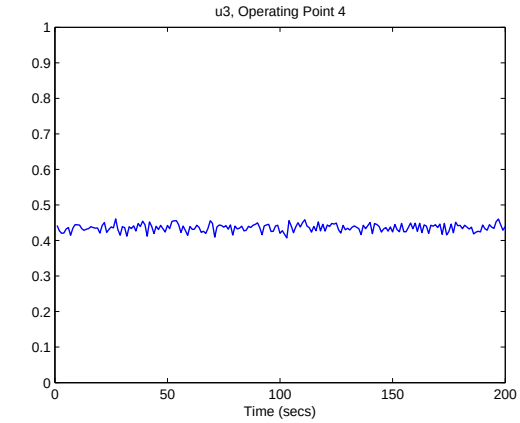

Figure 5.7: u_3 at Operating Point 4

5.6 Control of Boiler-Turbine System for Switching OPs

Now the case switching operating points is explored. The system is started at the nominal operating points in all cases. At t = 100s, the operating points are switched. The following switching cases are explored:

- Operating Points switch from OP#4 to OP#5.
- Operating Points switch from OP#4 to OP#6.
- Operating Points switch directly from OP#4 to OP#7.
- Operating Points switch from OP#4 to OP#3.
- Operating Points switch from OP#4 to OP#2.

The results for the first case are given in Figures 5.8 to 5.13. It is seen that the controller generates the optimal sequences needed to move the system states to the new operating points quite smoothly. As soon as the disturbance is applied, it is noted that the feed water is cut (u_3) to allow more steam to be produced from the existing water in the boiler. The flow of the fuel, u_1 is also increase as well as the steam flow, u_2. As soon as the system reaches the operating points, the control efforts are constant at their new values. It is clear that now more fuel and water is allowed into the boiler to generate more steam required to produce more power. The pressure in the boiler thus rises.

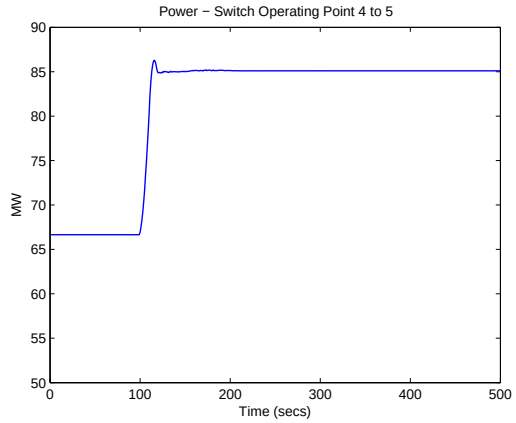

Figure 5.8: Power Generated - Switch Operating Points from 4 to 5

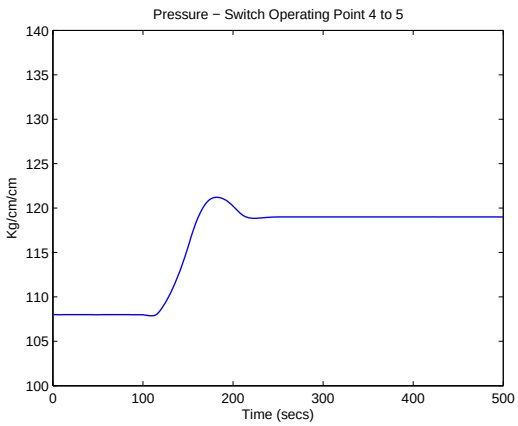

Figure 5.9: Pressure - Switch Operating Points from 4 to 5

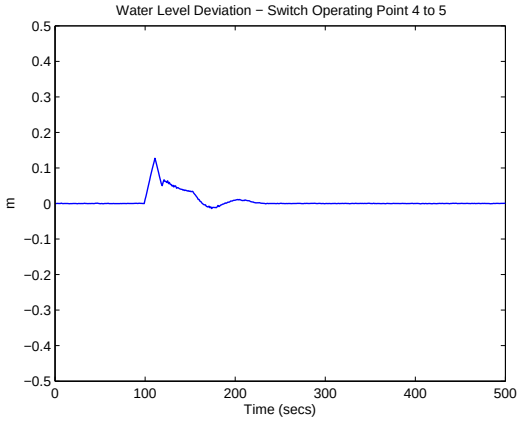

Figure 5.10: Water Level - Switch Operating Points from 4 to 5

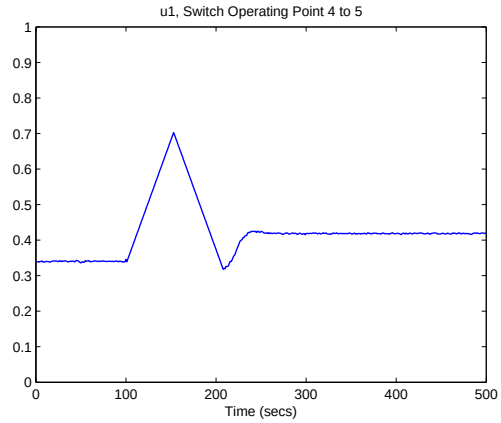

Figure 5.11: u_1 - Switch Operating Points from 4 to 5

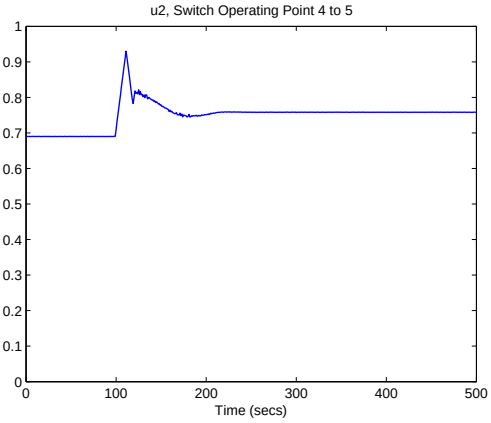

Figure 5.12: u_2 - Switch Operating Points from 4 to 5

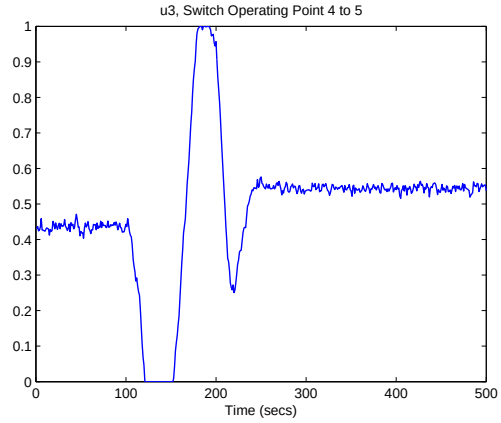

Figure 5.13: u_3 - Switch Operating Points from 4 to 5

It is seen that the power demand of 85.1 MW is met very early, merely 50s after the operating points are switched. The power output becomes completely steady after 100s. The maximum overshoot for the power is about 1.1 MW over the required operating value of 85.1 MW which appears for a few seconds. The power output reaches the new value so quickly because the cost function is weighted in such a way that it gives most cost to the error in power, compared to the error in pressure and water level deviation. This is very important because the power output is the most critical parameter and the power demand from the system should be fulfilled as quickly as possible.

The pressure reaches the steady operating value of 119 Kg/cm^2 after about 110s. The water level deviates to a maximum value of 9 cm during this switching. There are slight overshoots in the generated power and pressure of the boiler. This is is because of the slow response of the system and rate constraints on the control signals. It is seen that the pressure response is much slower compared to the output power response because the error in pressure is given a lesser weight.

For the second case, the same behavior is repeated, and the system has now to cope with a larger power demand. The results for this case seen in Figures 5.14 to 5.19. The system is able to reach the required output power value of 105 MW in less than 50s after the switching is done. This switch is smooth without any overshoots. The pressure takes about 200s to reach the steady value at this operating point. The value overshoots by about 6.2 Kg/cm^2 and then decreases to the steady value of 130 Kg/cm^2. The water level deviates to a maximum value of 15.6 cm in the drum.

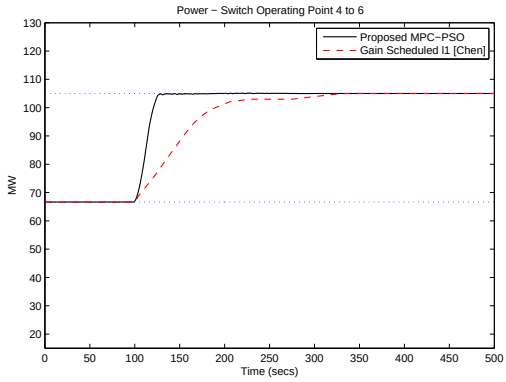

Figure 5.14: Power Generated - Switch Operating Points from 4 to 6

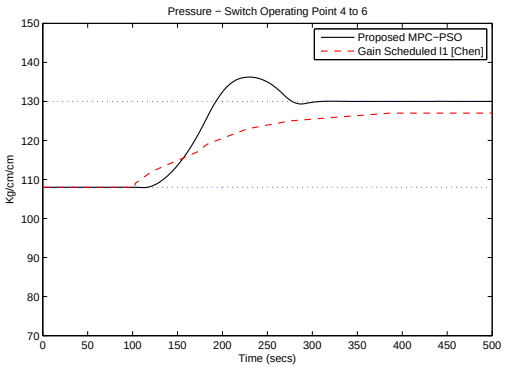

Figure 5.15: Pressure - Switch Operating Points from 4 to 6

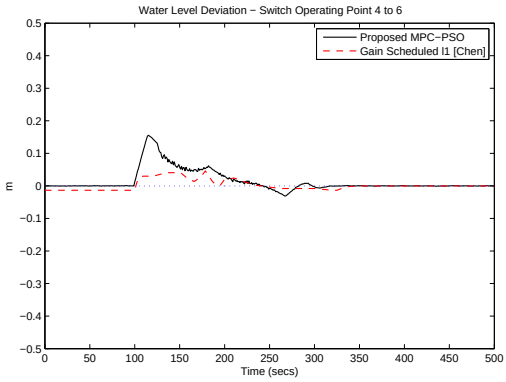

Figure 5.16: Water Level - Switch Operating Points from 4 to 6

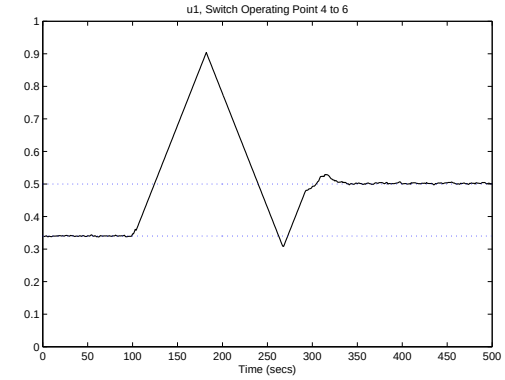

Figure 5.17: u_1 - Switch Operating Points from 4 to 6

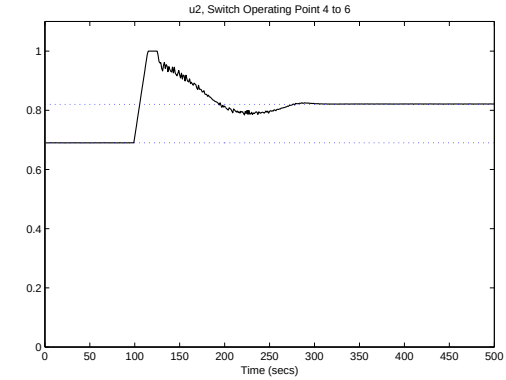

Figure 5.18: u_2 - Switch Operating Points from 4 to 6

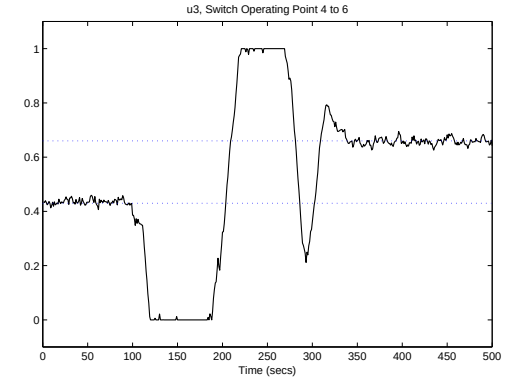

Figure 5.19: u_3 - Switch Operating Points from 4 to 6

This case is compared with the work done by Chen and Shamma [31]. It is found that the proposed control is much more successful on this complex system because it is able to fulfil the power demanded from it much quickly. Through the use of gain scheduled l^1-optimal control technique, Chen and Shamma are able to fulfil the power demand of 108 MW in about 125s, which is considerably more than the proposed technique. Also, the pressure is not able to reach the required steady state value of 130 Kg/cm^2 using their technique, but instead reaches only up to 127 Kg/cm^2. It is seen that the proposed controller is able to bring the pressure to the required equilibrium value. In this case the pressure reaches a maximum value of 136.2 Kg/cm^2, but this value is well within the allowable range of the system. The pressure increases because the focus is to provide the required power from this generation unit system as quickly as possible, as explained earlier. Water level deviation is slightly more in the proposed case because more water is needed to provide for the rise in pressure and quicker supply of electrical power.

It is seen in the second case that the system again reaches the new steady state values smoothly. It is also obvious that the constraints on the control efforts as well as the different rate constraints are obeyed during this transition. The rate change is maximum in the transient behavior of the system, and once the system reaches steady state, the control efforts are at constant values.

In the third case, the operation of the system is switched from operating point 4 to operating point 7 directly. This switching is not recommended, instead, for moving to operating point 7, it is recommended to switch to an intermediate operating point

like 5 or 6. However, for testing purposes this is done here. It is seen from Figures 5.20 to 5.25 that the system is able to switch its operation to the required operating point and fulfils the power demanded from it. It is observed that the power demand is fulfilled in about 193 seconds from the switching. The pressure takes longer, about 300 seconds to reach a steady state in this extreme case. A steady state error of about 1 Kg/cm^2 is observed in this case. The water level deviates to a maximum value of 15.9 cm for this case.

For the fourth and fifth cases, the power demanded from the turbine is decreased. The behavior of the system states is seen in Figures 5.26 to 5.31 and Figures 5.32 to 5.37. The supply of fuel and steam to the system is decreased as soon as the operating points are switched. This is done while obeying the rate constraints on the system given in Equation 5.7. To maintain the system at the new operating points, lower effort is needed which is clear in Figures 5.23 to 5.25. There are slight undershoots in these cases. However it is seen that the power demand is met within 50 s of the switching in both cases.

5.7 Conclusion

The proposed controller is able to switch the operating points of the complex system successfully. It is seen that the optimal control effort is applied allowing the system to reach its steady state values quickly and smoothly with minimum overshoots. The results show that the proposed controller can be directly applied to the nonlinear model of this coupled system. It generates optimal control signals for switching the

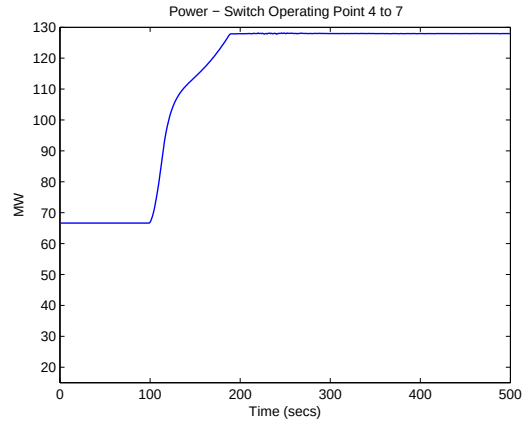

Figure 5.20: Power Generated - Switch Operating Points from 4 to 7

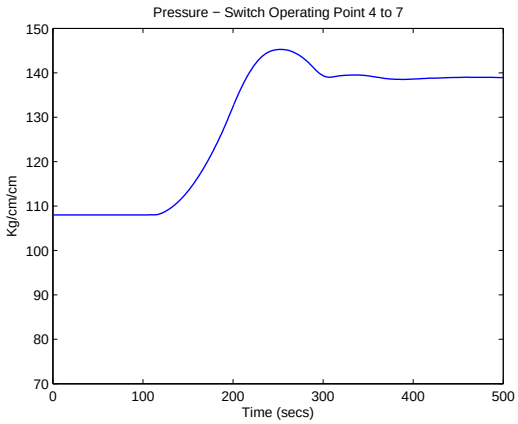

Figure 5.21: Pressure - Switch Operating Points from 4 to 7

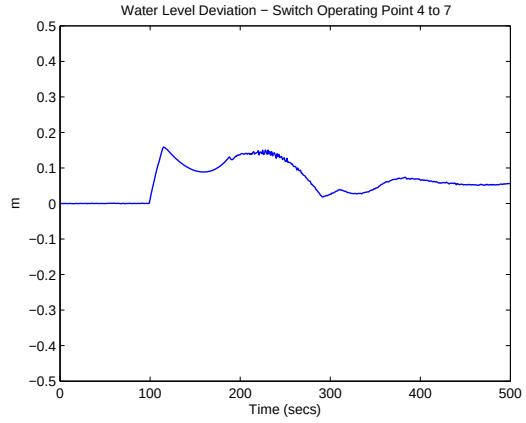

Figure 5.22: Water Level - Switch Operating Points from 4 to 7

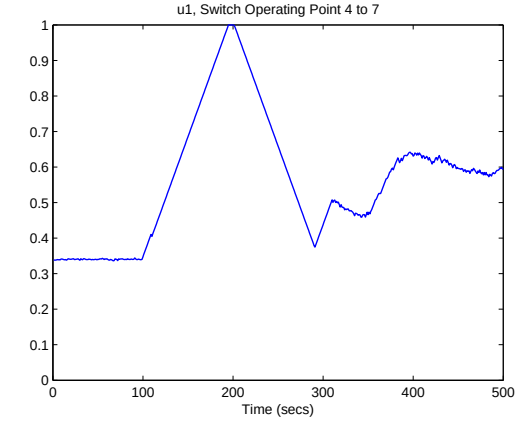

Figure 5.23: u_1 - Switch Operating Points from 4 to 7

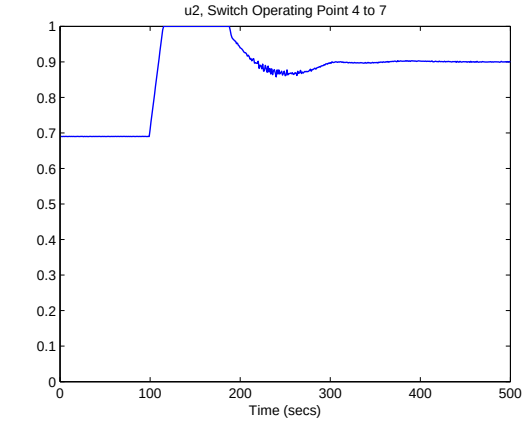

Figure 5.24: u_2 - Switch Operating Points from 4 to 7

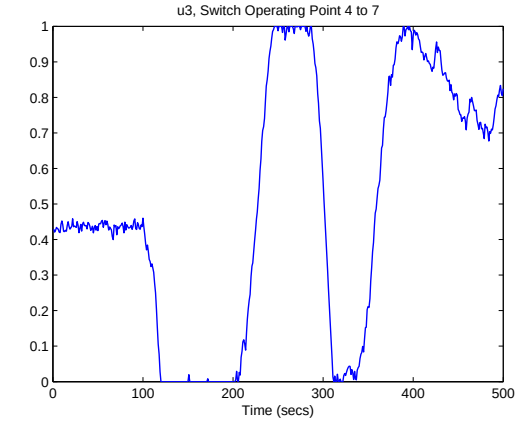

Figure 5.25: u_3 - Switch Operating Points from 4 to 7

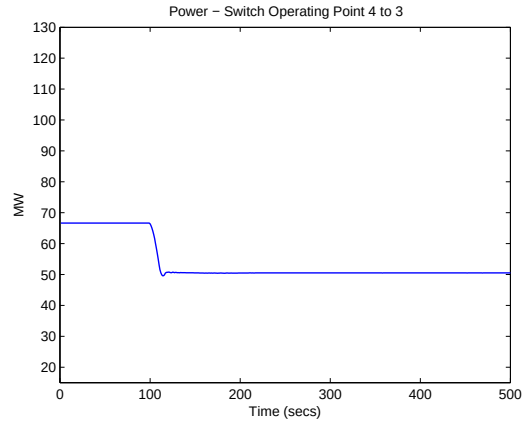

Figure 5.26: Power Generated - Switch Operating Points from 4 to 3

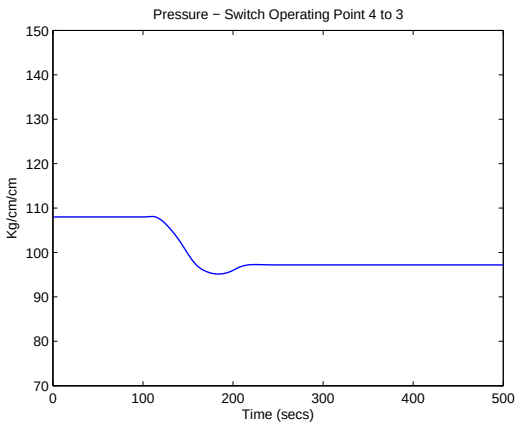

Figure 5.27: Pressure - Switch Operating Points from 4 to 3

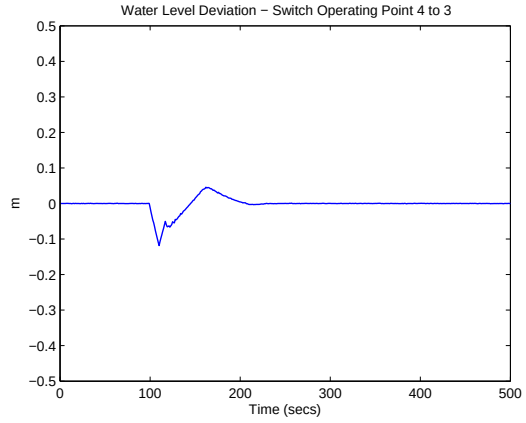

Figure 5.28: Water Level - Switch Operating Points from 4 to 3

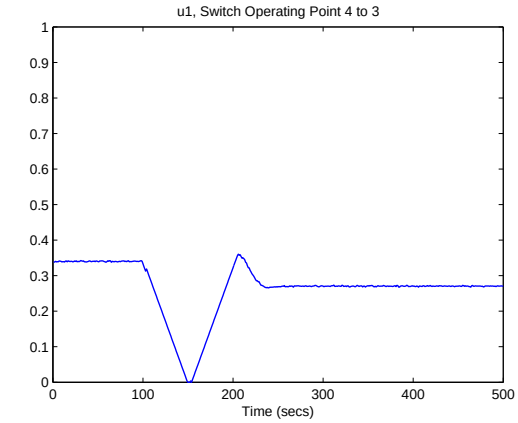

Figure 5.29: u_1 - Switch Operating Points from 4 to 3

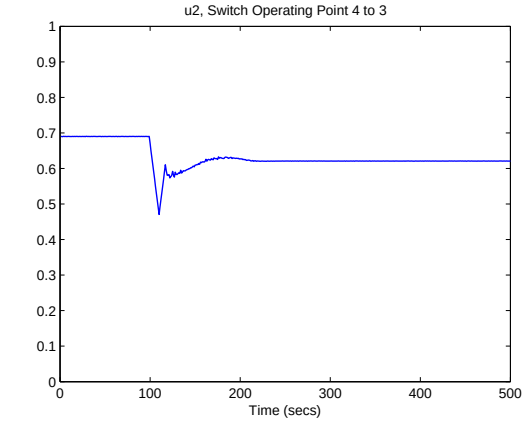

Figure 5.30: u_2 - Switch Operating Points from 4 to 3

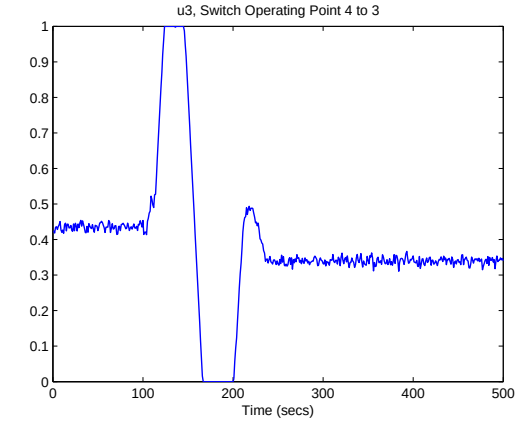

Figure 5.31: u_3 - Switch Operating Points from 4 to 3

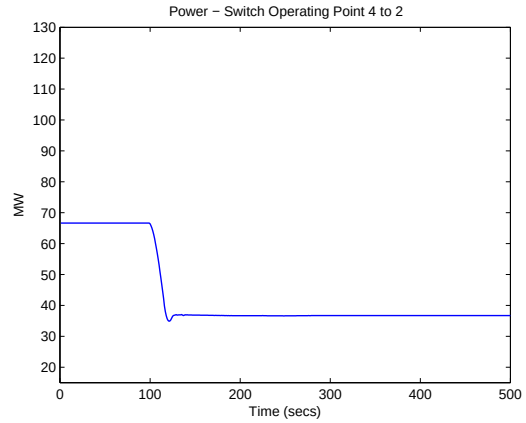

Figure 5.32: Power Generated - Switch Operating Points from 4 to 2

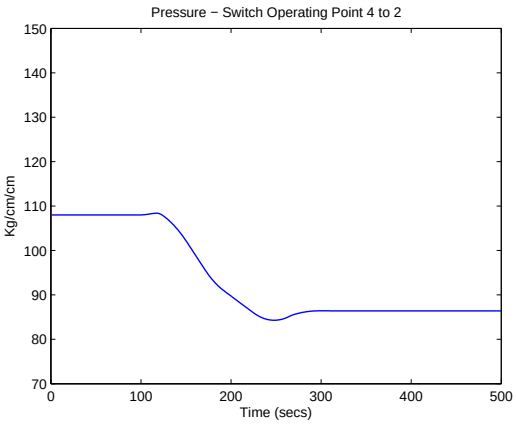

Figure 5.33: Pressure - Switch Operating Points from 4 to 2

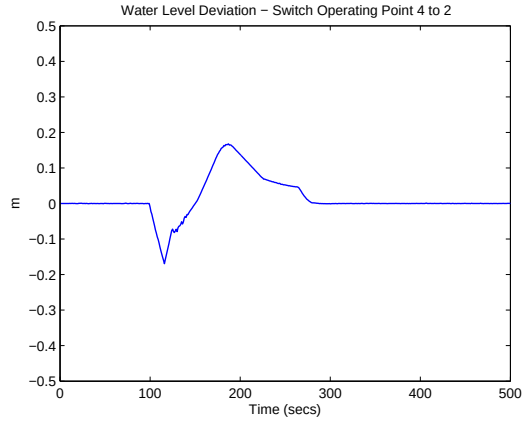

Figure 5.34: Water Level - Switch Operating Points from 4 to 2

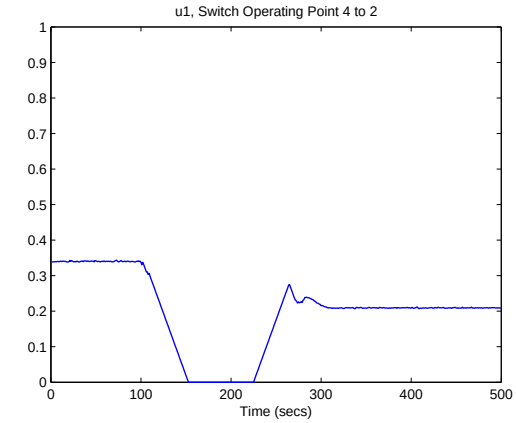

Figure 5.35: u_1 - Switch Operating Points from 4 to 2

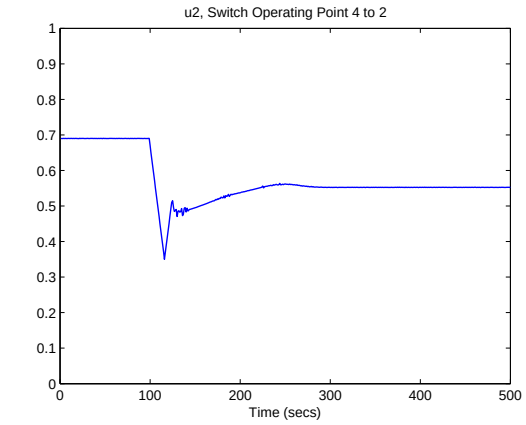

Figure 5.36: u_2 - Switch Operating Points from 4 to 2

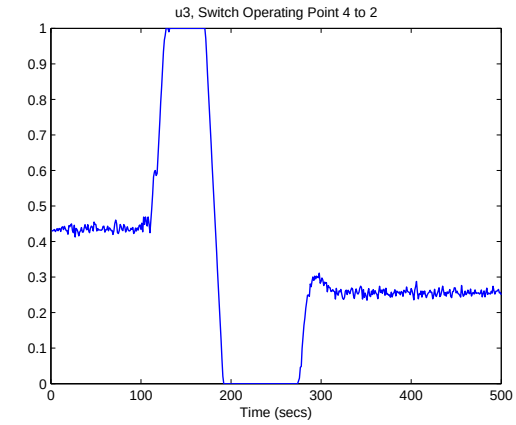

Figure 5.37: u_3 - Switch Operating Points from 4 to 2

plant from one operating point to another while obeying the limits of the control effort applicable, as well as applying the different rate limitations of the individual control efforts. Compared to previous work, the proposed technique provides faster control. An extreme case of switching operating points from nominal to maximum directly is also studied and it shows that the controller is able to cope up with this challenge and gives a smooth and quick transition. The controller is able to switch between different operating points of the system within the full range of operation effectively and demonstrates special emphasis on fulfilling the power demand as quickly and as smoothly as possible.

CHAPTER 6

CONCLUSIONS AND RECOMMENDATIONS

In this chapter some further topics related to the proposed MPC-PSO controller are discussed. A brief summary of this thesis is also presented. The conclusion derived from this work and recommendation for future work and improvements is also given.

6.1 Further Topics

6.1.1 MPC-PSO in Noisy Environment

Although all efforts to remove noise from a process are done, it is likely that the process will experience noise at one or more of its inputs. The propose technique is tested in such a case, when noise is present in the system. For this purpose, the nonlinear SMIB system, discussed in Chapter 3 is used. All parameters are same as already given with the same process constraints.

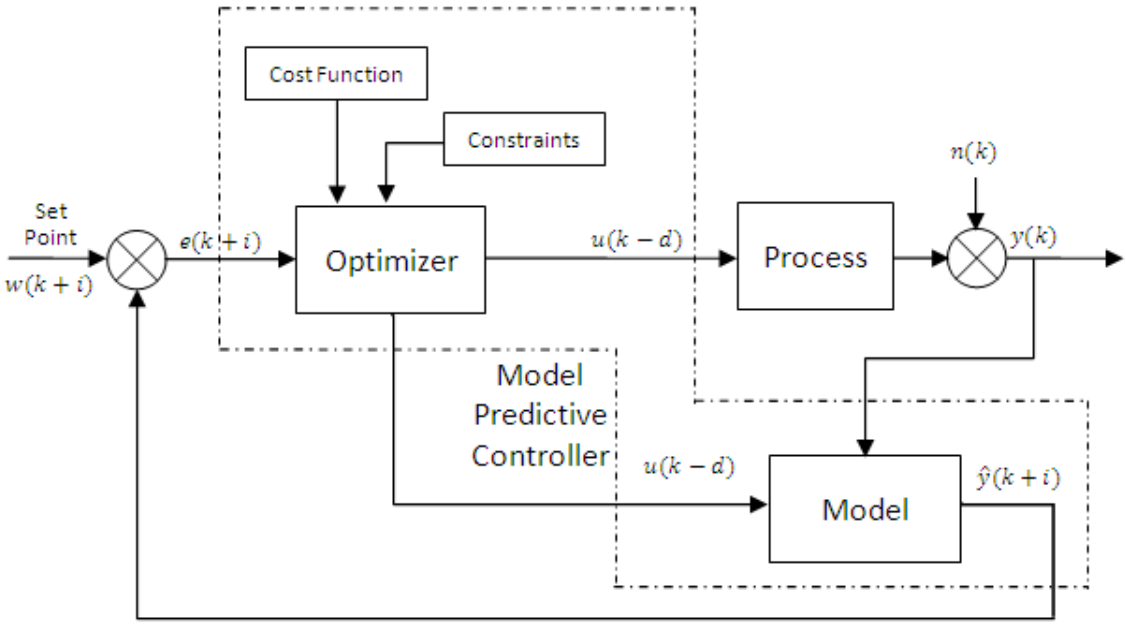

Figure 6.1: Block Diagram of Proposed MPC-PSO with Noise

The block diagram in Figure 6.1 shows the proposed controller with noise, $n(k)$ added at the feedback. This means that at the sampling of the output, a noise is present in the system and the signal that is eventually fed back to the controller is distorted. The noise signal is show in Figure 6.2 and it is clear that the system is subjected to a rather large random noise value, approaching that of the originally perturbed conditions of the signal, i.e. $|0.1|$ p.u. in magnitude. The magnitude of the noise signal is a minimum of 0.052 p.u.

The result of this simulation are seen in Figures 6.3 to 6.6. Although the system is subjected to a large noise signal, it is seen in Figure 6.4 that the there is very little continuous deviation in the frequency of the generated power. There are small fluctuations present in the control signal, u_2 due to the continuous random noise. However, the system is still able to track the required outputs, i.e. I_d and ω. In this

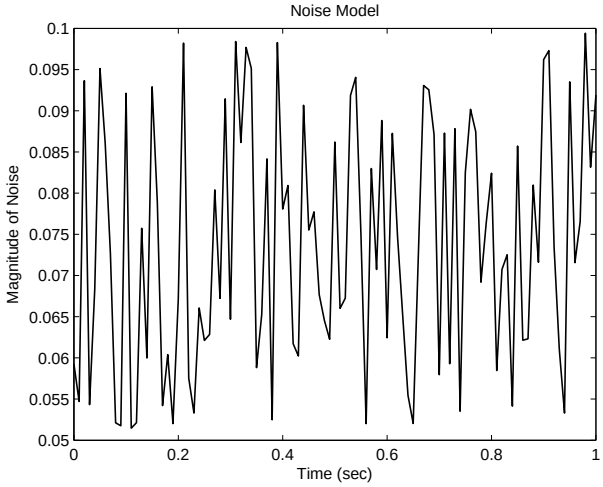

Figure 6.2: Noise in the System

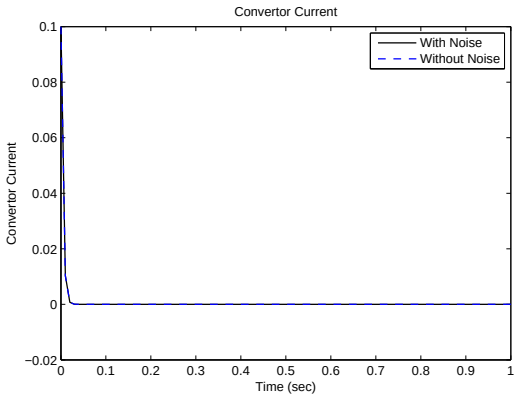

Figure 6.3: Converter Current for SMIB system with Noise

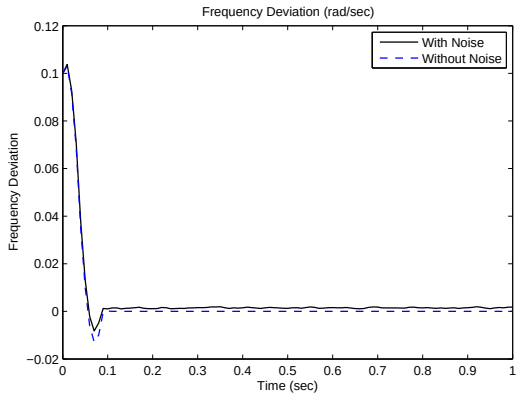

Figure 6.4: Frequency Deviation with Noise

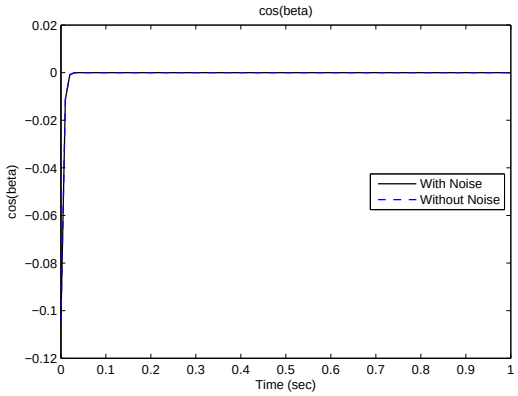

Figure 6.5: Control Effort, $cos(\beta)$ for Noise

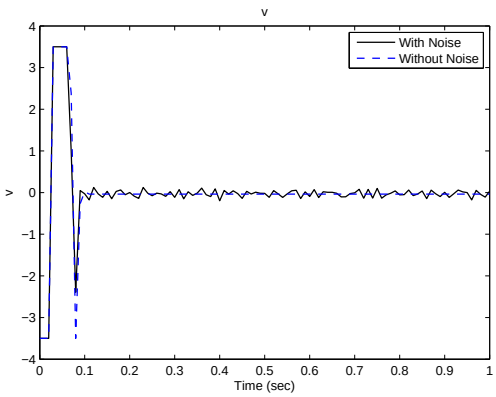

Figure 6.6: Control Effort, v for Noise

case, there is also an inherent risk that if the noise level at a particular instant is the negative of the output at that same instant, the output will become zero and control will become difficult.

6.1.2 MPC-PSO with Model Mismatch

It is unlikely that the explicit models of the complex processes we have studied are 100% accurate. In this section, the behavior of the proposed technique in the case of model mismatch is given. The SMIB example from Chapter 3 is again taken. However, now the parameters of the system have 5% modeling error. Thus, the parameters of the system are modeled as follows:

Table 6.1: Modeling Errors for SMIB Parameters

Parameter	Actual Value p.u.	Modeled Value p.u.
X	0.2	0.21
R_c	0.3	0.3150
L	0.015	0.0158

This in turn affects the system parameters,

$k_1 = \frac{R_c}{L}$, $k_2 = \frac{\omega_B E_1 E_2}{2HX}$, $k_3 = \frac{\omega_b R_c}{2HX}$, $k_4 = \frac{1}{L}$, $k_5 = \frac{\omega_B}{2H}$

Taking $H = 7.0\ s$, $D = 0.5\ s^{-1}$, and α = -0.1 s^{-1}, this corresponds to:

Table 6.2: Effects of Modeling Errors on System Parameters

Parameter	Actual Value p.u.	Modeled Value p.u.
k_1	20	19.9367
k_2	177.72857	169.2652
k_3	8.078571	8.0786
k_4	66.667	63.2911
k_5	26.928561	26.9286

As shown in Figures 6.7 to 6.10, there is a steady state error in the output due to the modeling error. The output frequency of the system is deviated by 0.01 rad/s which is understandable as the model is different from the system.

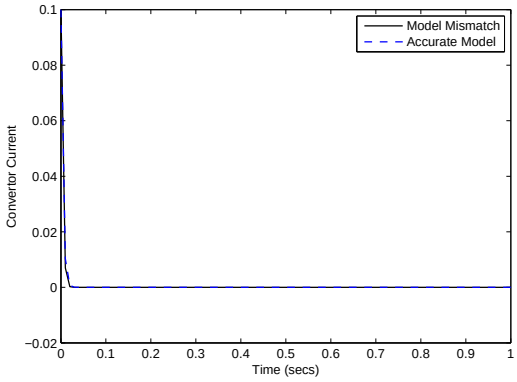

Figure 6.7: Converter Current for SMIB system with Model Mismatch

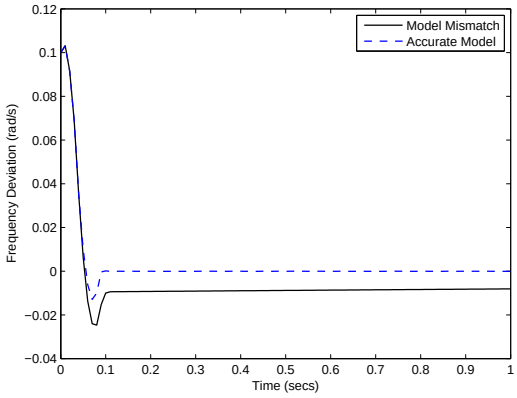

Figure 6.8: Frequency Deviation with Model Mismatch

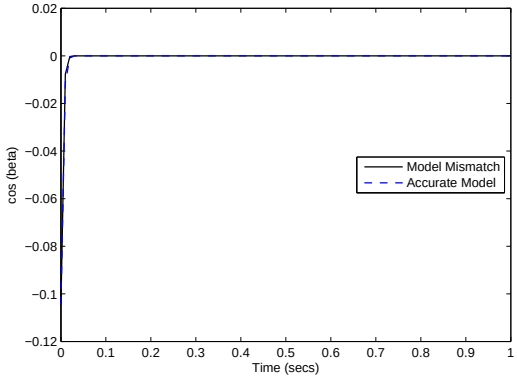

Figure 6.9: Control Effort, $cos(\beta)$ for Model Mismatch

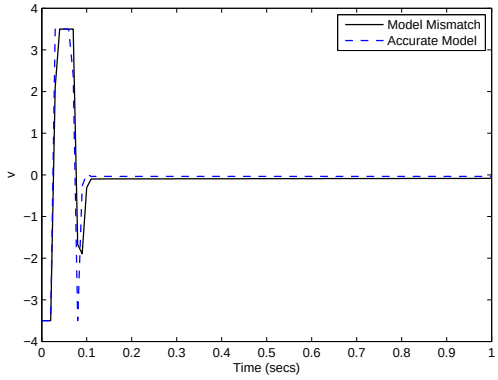

Figure 6.10: Control Effort, v for Model Mismatch

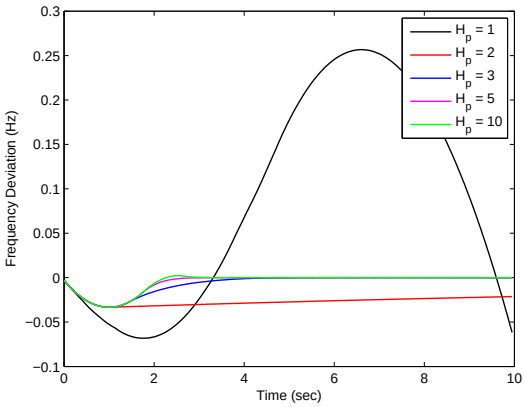

Figure 6.11: Effect of Prediction Horizon - Change in Frequency Deviation

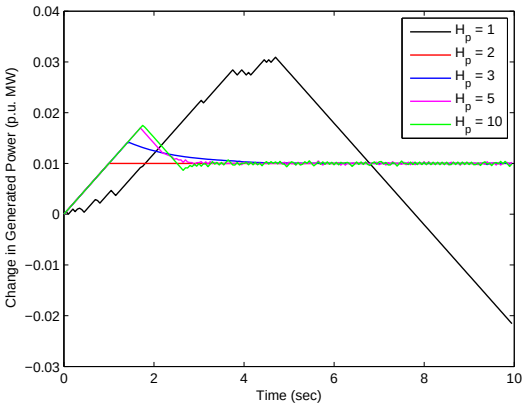

Figure 6.12: Effect of Prediction Horizon - Change in Power Generated

6.1.3 Effect of Prediction Horizon

The Prediction Horizon, H_p is one of the most important parameters of MPC and is used to tune the design controller. Increasing H_p generally has a stabilizing effect on the closed loop system, and typically results in better control. However, due to large H_p, the computational load on the system increases considerably. This load comes from the least-squares cost functions that have been defined previously. Therefore, the computational price and performance must be balanced through selecting an appropriate value of H_p. So, for robustness purposes, large value of H_p is advisable, but for computational purposes, a smaller value is preferred. It is also argued that the H_p should be chosen such that the system reaches 90% of its steady state value within the horizon [115]. In this thesis, the H_p for which the best results were obtained was selected for simulation.

Figure 6.11 shows the simulation of the Load Frequency Control problem for Single Area with GRC value = 0.01 p.u. MW/min. It is seen that for the case when H_p

= 1, the system is totally uncontrollable. This is because an $H_p = 1$ is less than the actual delay of the system, and the MPC is not able to control it. It is noted that the minimum H_p should be at least equal to the delay in the system. So, for H_p = 2, it is seen that although the frequency deviation parameter is converging to the required value of zero, the response is very sluggish. As H_p increases to 3 and 5, the response improves and the frequency deviation becomes zero quickly. However, as seen in Figure 6.12, due to large value of H_p, the computational load significantly increases, hence sub-optimal control sequences are generated which although bring the frequency deviation to zero, cause small fluctuations in the power deviation. Therefore, from an overall perspective of the system, it appears that selecting $H_p = 3$ gives the best results.

6.1.4 Effect of PSO Population Size

The population size of the particle swarm for the PSO technique also affects the quality of the optimal control given by the proposed technique. This is because the population of the swarm is the number of solutions available. The larger the swarm, the larger will be the pool of potentially optimal solutions that will give the lowest cost.

Keeping all other parameters same, and an H_p of 3, the single area nonlinear LFC system is simulated with different sizes of the particle swarm. As seen in Figure 6.13, it is seen that the frequency deviation is controlled to be zero in exactly the same way, regardless of the swarm being large or small. However, from Figure 6.14, it is

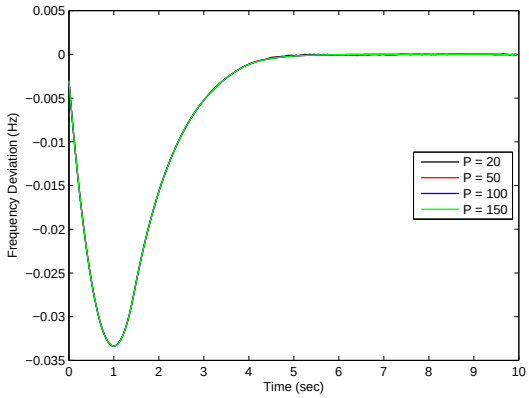

Figure 6.13: Effect of Population Size - Change in Frequency Deviation

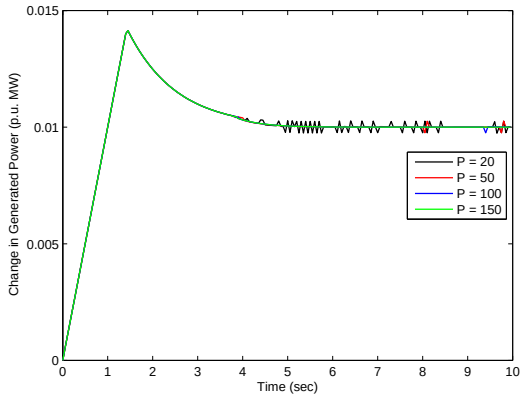

Figure 6.14: Effect of Population Size - Change in Power Generated

seen that when the population of the swarm is small, at 20, there are fluctuations in the power generated from the system, which is again the result of sub-optimal control signals generated due to small swarm size. As the population is increased, the power generated becomes smooth, with essentially no difference in the results if the population size is increase beyond 100 particles.

This observation is interesting because it also illustrates that even though the controlled variable may come to the required steady state, the parameters of the proposed controller like the population size and H_p can affect the overall system in such a way that other outputs, which are not controlled, may experience fluctuations. Therefore, the controller has to be tuned keeping in view the complete system and what is required from it.

6.1.5 Computational Complexity and Real-Time Application

There are a number of factors that determine the computational complexity of the proposed technique. The main factors contributing to the computational complexity

are:

1. Size of Control Loop
2. Complexity of the Process itself
3. Prediction Horizon, H_p
4. Population size of the particle swarm
5. Number of iterations of PSO

To keep the computational complexity low, limits are imposed on the sizes of each of the above factors. However, usually the techniques given in literature are computationally very complex and impossible to realize on hardware that can control the system in real-time.

A great advantage of the proposed technique is that it is extremely fast, compared to other techniques in the literature. This advantage mainly comes from the incorporation of PSO, instead of other optimization techniques. PSO is an inherently fast technique and it complements the efficient MPC in a nice way. Therefore, the computation time for a case when the population size is 20, H_p is 3, and number of iterations is 500 is merely 0.15s per sample on an Intel Core2Duo 2Ghz machine running Windows XP.

This time can be drastically improved by optimizing the coding and implementing the controller on a specialized industrial grade microprocessor. A wide variety of PIC-family controllers, and digital signal processors can be used for this purpose. If the computational complexity is further reduced, and sufficient hardware capability is

provided, the proposed technique can be realized comfortably for real-time application. Another solution is to realize the proposed technique on a cluster of PCs.

6.2 Summary

The thesis can be summarized by the following brief points:

- A new Model Predictive Control technique based on the Particle Swarm Optimization algorithm is proposed. The PSO algorithm is used as an optimization tool for the well-known MPC methodology. The proposed controller is called the MPC-PSO controller.

- Useful literature reviews to both MPC and PSO techniques are given in this thesis. It is tried that a concise yet comprehensive literature review is presented.

- The proposed control technique is applied for the optimal control of power systems. MPC techniques are mostly applied on processes in the existing literature and power systems are seldom controlled by MPC. The proposed technique is applied to three well-known, hugely practical and important power systems which are:

    1. Synchronous Machine on Infinite Bus (SMIB)

    2. Load Frequency Control (LFC) of multi-area interconnected power systems

    3. Fossil Fuel Power Unit - Drum-Boiler-Turbine system

- The proposed technique is fairly generic to any type of nonlinear constrained or

unconstrained system. It can be applied to any process, be it of power, chemical or industrial nature, with very few and minor modifications based on the system dynamics and constraints.

- The PSO algorithm incorporated in the MPC technique enables the controller to keep computational time low and arriving at optimal solutions of the control effort quickly.

- Explicit models of these power systems are considered. Nonlinearities are incorporate directly without any transformation, approximation, model reduction, linearization etc.

- Linear and non-linear both LFC are considered.

- SISO as well as MIMO systems are considered.

- The proposed technique is applied to the control of an automatic flow valve also, to illustrate the method and give a clearer understanding of the technique proposed.

- The thesis only deals with the control of the systems, with no effort for identification. Identification can also be incorporated with this control technique.

- Process and input constraints are explicitly taken into account in the system model and controller formulation during the optimization. Equality and rate constraints are incorporated wherever defined by the system dynamics.

- Wherever appropriate, the results obtained by the use of this proposed technique are compared with the results of various techniques present in the existing literature.

- The results are given such that a clear description of the dynamics of the relevant system states and control efforts are clear from the figures with the need of a lot of explanation.

- Special topics pertaining to the proposed controller are discussed.

6.3 Conclusion

The main things that can be concluded from this thesis are:

- Most of the work done on the control of power systems as well as processes using MPC uses linearized, approximated, reduced or transformed models of the system. The dynamics of the system are usually linearized around certain operating points. When setpoints rapidly change, the linearization techniques do not perform well. Even with good approximations and linearization techniques, there is always the drawback of inaccuracy, thus resulting in sub-optimal control. The proposed method provides a valuable technique that can be directly applied to nonlinear models of processes, with the confidence that the nonlinearities and constraints of the system will be dealt with efficiently and optimal control signals will be generated.

- The only limitation to the proposed technique is due to the physical limitations

of the actuators being controlled. Therefore, the proposed technique can continuously perform better due to the fact that faster, more accurate and reliable actuators are developed continuously.

- Input and rate constraints can be effectively imposed on the controller by design, thus ensuring that the constraints are never violated.

- The application of the proposed technique on SMIB system show that the controller performs extremely well for the cases of perturbed initial conditions as well as for induced faults and parameter variations. The overshoots in the system states as a result of these perturbations and faults were minimal and the control effort applied was smooth and quick. A comparison of the results obtained with preceding literature showed that the proposed controller performs far better than variable structured control for this case.

- Results for the application of the proposed technique to the LFC system show that the system is well-suited to take on this challenging problem. The results show that the controller is able to successfully control the load-frequency of single as well as interconnected power systems better than what was achieved in previous work on this topic.

- The proposed controller is able to switch the operating points of the complex fossil fuel power unit quite successfully. Optimal control efforts are applied allowing the system to reach the required steady state values quickly and smoothly with minimum overshoots. The results show that the proposed controller can be

directly applied to the nonlinear model of this highly nonlinear coupled system. The controller always obeys the limits of the control effort applicable, as well as applies the different rate limitations of the individual control efforts. It is seen that controller is able to switch between different operating points of the system within the full range of operation effectively and demonstrates special emphasis on fulfilling the power demand as quickly and as smoothly as possible.

6.4 Recommendation for Further Work

Research is always considered as an iterative as well as exhaustive process. The work on the proposed controller and this topic, by a true measure, can not be considered complete. There will always be points to improve upon and more problems to tackle. This, when accompanied by the eagerness to excel in this field, gives rise to several recommendations. Rest assured, ample time and energy will be invested to further work on this vast topic and positively contribute to science. The many possible suggestions that come to mind in this regard are as follows:

- Several other EAs and optimization techniques remain to be explore as a tool for MPC. A variety of heuristics-based MPC techniques can be developed and compared for different problems.

- The proposed technique uses the PSO tool in its relatively simplest form. Several advanced as well as hybrid PSO techniques are abundantly available in the literature. The performance of the proposed controller can be improved by incorporating an advanced PSO algorithm.

- An important factor in the performance of the predictive controller is the nature of the cost function. The cost function can be formulated in several ways, and can include other system parameters along with the error. Therefore, performance of the proposed controller using different cost functions (depending on the application) can be investigated.

- Only power systems are considered. This technique will undoubtedly find numerous applications in the field of petrochemical and industrial processes. For sure, it will provide valuable contribution to the optimal control of processes.

- The proposed technique can be applied to a multi-area SMIB system in which several synchronous machines are connected to an infinite bus.

- More ideas related to the PSO and MPC parameters can be explored.

- This technique coupled with a good system identification technique can provide a powerful package for identification and control of complex nonlinear systems.

- Adaptive MPC-PSO methodology can be developed. This methodology will then be used for the online control of a process whose model is uncertain. The adaptive MPC-PSO will be able to identify this mismatching online and update the process model such that it is accurate and optimal control can be achieved.

- Efforts can be done for reducing the computational complexity of this algorithm. This will be a tremendous help in applying the system to complex systems in real-time using off-the-shelf hardware.

- Hardware Realizability and Real-time Implementation of the proposed technique are very interesting and promising topics which deserve to be further investigated.

REFERENCES

[1] Åström KL and Eklund K. A simplified nonlinear model of a drum-boiler-turbine unit. *International Journal of Control*, 16(1):145–169, 1972.

[2] Al-Duwaish HN and Naeem W. Nonlinear model predictive control of hammerstein and wiener models using genetic algorithms. *Proceedings of the IEEE International Conference on Control Applications*, pages 465–469, 2001.

[3] Al-Hamouz ZM and Al-Duwaish HN. A new load frequency variable structure controller using genetic algorithms. *Electric Power Systems Research*, 55(1):1–6, 2000.

[4] Al-Musabi NA. Design of optimal variable structure controllers: Applications to power system dynamics. *MS Thesis, King Fahd University of Petroleum & Minerals Dhahran, KSA*, 2004.

[5] Alamir M, Sheibat-Othman N, and Othman S. Constrained nonlinear predictive control for maximizing production in polymerization processes. *IEEE Transactions on Control Systems Technology*, 15(2):315–323, 2007.

[6] Allgower F. Analysis and controller synthesis for nonlinear processes using nonlinearity measures. *AICHE Annual Meeting No. 180h*, 1995.

[7] Amjad S and Al-Duwaish HN. Model predictive control of shell benchmark process. *Proceedings of the 10th IEEE International Conference on Electronics, Circuits and Systems*, 21:655–658, 2003.

[8] Angeline PJ. Evolutionary optimization versus particle swarm optimization: Philosophy and performance differences. *Lecture Notes in Computer Science*, 1447:601–610, 1998. Book Title: Evolutionary Programming VII, ISBN 978-3-540-64891-8.

[9] Bacic M, Cannon M, and Kouvaritakis B. Invariant sets for feedback linearisation based nonlinear predictive control. *IEE Proceedings of Control Theory and Applications*, 152(3):259–265, 2005.

[10] Balan R, Maties V, Hancu O, and Stan S. A model predictive control algorithm applied to nonlinear processes. *PAMM Proceedings of Applied Maths & Mechanics*, 6(1):797–798, 2006.

[11] Balbis L, Katebi R, and Dunia R. Graphical based predictive control design. *Proceedings of the 2005 IEEE Conference on Control Applications*, pages 297–302, 2005.

[12] Balbis L, Katebi R, Dunia R, Ordys A, and Grimble MJ. Nonlinear predictive control for real time applications. *Proceedings of the 2006 IEEE International Conference on Control Applications*, pages 211–216, 2006.

[13] Bell RD and Åström KL. Dynamic models for boiler-trubine alternator units: Data logs and parameters estimation for 160mw unit. *Lund Institute of Technology, Technical Report*, 1987.

[14] Bevrani H, Mitani Y, and Tsuji K. Sequential design of decentralized load frequency controllers using synthesis and analysis. *Journal of Energy Conversion and Management*, 45(6):865–881, 2004.

[15] Birch AP, Sapeluk AT, and Özveren CS. An enhanced neural network load frequency control technique. *IEE International Conference on Control*, 389:409–415, 1994.

[16] Bloemen HHJ, van den Boom TJJ, and Verbruggen HB. Model-based predictive control for hammerstein-wiener systems. *International Journal of Control*, 74(5):482–495, 2001.

[17] Botto MA and da Costa JS. A comparison of nonlinear predictive control techniques using neural network models. *International Workshop on Neural Networks for Identification, Control, Robotics, and Signal/Image Processing (NICROSP '96)*, page 419, 1996.

[18] Bouani F, Laabidi K, and Ksouri M. Constrained nonlinear multi-objective predictive control. *IMACS Multiconference on Computational Engineering in Systems Applications (CESA)*, pages 1558–1565, 2006.

[19] Brooms AC, Cannon M, Kouvaritakis B, and Lee YL. Efficient nonlinear model predictive control. *Proceedings of the American Control Conference*, pages 255–259, 2000.

[20] Bulut B, Ordys AW, and Grimble MJ. Application of efficient nonlinear predictive control to a hot strip finishing mill. *Proceedings of the 2002 IEEE International Conference on Control Applications*, 1:373–378, 2002.

[21] Calovic M. Linear regulator design for load and frequency control. *IEEE Winter Power Meeting*, 1971.

[22] Cam E and Kocaarslan I. Load frequency control in two area power system using fuzzy logic controller. *Journal of Energy Conversion and Management*, 45:233–245, 2005.

[23] Cannon M, Kouvaritakis B, Lee YI, and Brooms AC. Efficient nonlinear model based predictive control. *International Journal of Control*, 74(4):361–372, 2001.

[24] Cao Y, Jiang L, Cheng S, Chen D, Malik OP, and Hope GS. A nonlinear variable structure stabilizer for power system stability. *IEEE Transactions on Energy Conversion*, 9:489–495, 1994.

[25] Carlisle A and Dozier G. An off-the-shelf pso. *Proceedings of the Particle Swarm Optimization Workshop*, pages 1–6, 2001.

[26] Cavin R, Budge M, and Rasmussen P. An optimal linear systems approach to load frequency control. *IEEE Winter Power Meeting*, 1970.

[27] Chan WC and Hsu YY. Automatic generation control of interconnected power system using variable structure controllers. *Proceedings of the IEE Part C*, 128(5):269–279, 1981.

[28] Chang J, Lee KY, and Garduno-Ramirez T. Multiagent control system for a fossil-fuel power unit. *IEEE Power Engineering Society General Meeting*, 3:1465, 2003.

[29] Chemachema M, Megri F, and Belarbi K. A fuzzy dynamic programming solution to constrained nonlinear predictive control. *IEEE International Conference on Systems, Man and Cybernetics*, 5, 2002.

[30] Chemori A and Alamir M. Generation of multi-steps limit cycles for rabbit using a low dimensional nonlinear predictive control scheme. *Proceedings of 2004 International Conference on Intelligent Robots and Systems*, 3:2259–2264, 2004.

[31] Chen PC and Shamma JS. Gain scheduled l^1-optimal control for boiler-turbine dynamics with actuator saturation. *Journal of Process Control*, 14:263–277, 2003.

[32] Cheng X and Krogh BH. Stability constrained model predictive control for nonlinear systems. *Proceedings of the 36th Conference on Decision and Control*, pages 2091–2096, 1997.

[33] Choi SS, Sim HK, and Tan KS. Load frequency control via constant limited-state feedback. *Electric Power Systems Research*, 4(4):265–269, 1981.

[34] Clarke D, Ed. *Advances in Model-Based Predictive Control*. Oxford University Press, 1994.

[35] Clerc M. The swarm and the queen: Towards a deterministic and adaptive particle swarm optimization. *Proceedings of the 1999 Congress on Evolutionary Computation*, pages 1951–1957, 1999.

[36] Cortes P, Rodreguez J, Vargas R, and Ammann U. Cost function-based predictive control for power converters. *32nd Annual Conference on IEEE Industrial Electronics*, pages 2268–2273, 2006.

[37] Cutler JC and Ramaker B. Dynamic matrix control - a computer control algorithm. *Proceedings of the Joint Automatic Control Conference*, 1980.

[38] Das D, Kothari ML, Kothari DP, and Nanda J. Variable structure control strategy to automatic generation control of interconnected reheat thermal systems. *IEE Proceedings on Control Theory and Applications*, 138(6):579–585, 1991.

[39] Demello FP and Concordia C. Concepts of synchronous machine stability as affected by excitation control. *IEEE Transactions on Power Apparatus and Systems*, 88:316–329, 1969.

[40] deOliveira SL. Model predictive control for constrained nonlinear systems. *PhD Thesis, California Institute of Technology (Caltech)*, 1996.

[41] Deshpande PB, Ed. *Multivariable Process Control*. Instrument Society of America, 1989.

[42] Dimeo R and Lee KY. Boiler-turbine control system design using a genetic algorithm. *IEEE Transactions on Energy Conservation*, 10(4):752–759, 1995.

[43] Naidu DS. *Optimal Control Systems*. CRC Press, Boca Raton, FL, 2003.

[44] Dutka AS, Ordys AW, and Grimble MJ. Non-linear predictive control of 2 dof helicopter model. *Proceedings of the 42nd IEEE Conference on Decision and Control*, 4:3954–3959, 2003.

[45] Eberhart R and Shi Y. Comparing inertia weights and constriction factors in particle swarm optimization. *Proceedings of the Congress on Evolutionary Computing*, pages 84–89, 2000.

[46] Eberhart R and Shi Y. Particle swarm optimization: developments, applications and resources. *Proceedings of the 2001 Congress on Evolutionary Computation*, 1:81–86, 2001.

[47] El-Gallad A, El-Hawary M, Sallam A, and Kalas A. Enhancing the pso via proper parameters selection. *Proceedings of the IEEE Canadian Conference on Electrical and Computer Engineering*, 2:792–797, 2002.

[48] Espinosa J and van de Walle J. Nonlinear predictive control using fuzzy models and semidefinite programming. *18th International Conference of the North American Fuzzy Information Processing Society*, pages 174–178, 1999.

[49] Fischer M. Analysis of nonlinear predictive control with extended dynamic matrix control. *Proceedings of the 2006 IEEE International Conference on Control Applications*, pages 966–971, 2006.

[50] Fogel LJ, Zurada JM, Marks II RJ, and Robinson CJ. Evolutionary programming in perspective: The top-down view. *Computational Intellingece: Imitating Life*, pages 135–146, 1994. IEEE Press.

[51] Garduno-Ramirez R. Overall intelligent hybrid control system for a fossil-fuel power unit. *PhD Dissertation, The Pennsylvania State University*, 2000.

[52] Geng F and Zhu X. A guidance law for multipurpose unmanned attack air vehicle based on the nonlinear predictive control. *Proceedings of the IEEE International Conference on Mechatronics and Automation*, pages 1501–1507, 2007.

[53] Gerksic S, Juricic D, Strmcnik S, and Matko M. Adaptive implementation of wiener model based nonlinear predictive control. *Proceedings of the IEEE International Symposium on Industrial Electronics*, 3:1159–1164, 1999.

[54] Ghosh A, Ledwich G, Malik OP, and Hope GS. Power system stabilizer based on adaptive control techniques. *IEEE Transactions on Power Apparatus and Systems*, 103(8):1983–1989, 1984.

[55] Ghosh SP. Optimizations of pid gains by particle swarm optimizations in fuzzy based automatic generation control. *Electric Power Systems Research*, 72(3):203–212, 2004.

[56] Goldberg DE. *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison-Wesley Longman Publishing Co., Boston MA, 1989.

[57] Halldorsson U, Fikar M, and Unbehauen H. Multirate nonlinear predictive control. *Proceedings of the American Control Conference*, 5:4191–4196, 2002.

[58] Halldorsson U, Fikar M, and Unbehauen H. Nonlinear predictive control with multirate optimisation step lengths. *IEE Proceedings on Control Theory Applications*, 152(3):273–284, 2005.

[59] Hameida AM. Wavelet neural network load frequency controller. *Journal of Energy Conversion and Management*, 46:1613–1630, 2005.

[60] Hedjar R, Toumi R, Boucher P, and Dumur D. Cascaded nonlinear predictive control of induction motor. *Proceedings of the 2000 IEEE International Conference on Control Applications*, pages 698–703, 2000.

[61] Hedjar R, Toumi R, Boucher P, and Dumur D. Feedback nonlinear predictive control of rigid link robot manipulators. *Proceedings of the 2002 IEEE International Conference on Control Applications*, 5:3594–3599, 2002.

[62] Hedjar R, Toumi R, Boucher P, and Dumur D. Two cascaded nonlinear predictive controls of induction motor. *Proceedings of the 2003 IEEE International Conference on Control Applications*, 1:458–463, 2003.

[63] Hedjar R, Toumi R, Boucher P, and Dumur D. End point constraints nonlinear predictive control with integral action for induction motor. *Proceedings of the 2004 IEEE International Conference on Control Applications*, 2:1691–1696, 2004.

[64] Hedjar R, Toumi R, Boucher P, Dumur D, and Tebbani S. Finite horizon nonlinear predictive control with integral action of rigid link manipulators. *Proceedings*

of the 2005 IEEE International Conference on Control Applications, 5:283–288, 2005.

[65] Heo JS and Lee KY. Multiobjective control of power plants using particle swarm optimization techniques. *IEEE Transactions on Energy Conversion*, 21(2):552–561, 2006.

[66] Hiyama T and Sameshima T. Fuzzy logic control scheme for online stabilization of multimachine power system. *Fuzzy Sets Systems*, 39:181–194, 1991.

[67] Hogg BW and El-Rabaie NM. Multivariable generalized predictive control of a boiler system. *IEEE Transactions on Energy Conservation*, pages 282–288, 1991.

[68] Hu K and Yuan J. Multi-model predictive control method for nuclear steam generator water level. *Journal of Energy Conversion and Management*, 49(5):1167–1174, 2008.

[69] Hunt LR, Su R, and Meyer G. Global transformations of nonlinear systems. *IEEE Transactions on Automatic Control*, 28(2):704–707, 1983.

[70] Imsland L, Bar N, and Foss BA. More efficient predictive control. *Automatica*, 41(8):1395–1403, 2005.

[71] Ionescu C, Zlate S, and De Keyser R. An approach to control the blood glucose level in diabetic patients. *IEEE International Conference on Automation, Quality and Testing*, 1:48–53, 2006.

[72] Iplikci S. Support vector machines-based generalized predictive control. *International Journal of Robust Nonlinear Control*, 16(17):843–862, 2006.

[73] Rossiter JA. *Model-Based Predictive Control - A Practical Approach.* CRC Press, 2003.

[74] Maciejowski JM. *Predictive Control with Constraints.* Pearson Education Limited, 2002.

[75] Jurado F and Carpio J. Improving distribution system stability by predictive control of gas turbines. *Journal of Energy Conversion and Management*, 47(18-19):2961–2973, 2006.

[76] Katare S, Kalos A, and West D. A hybrid swarm optimizer for efficient parameter estimation. *Congress on Evolutionary Computation*, 1:309–315, 2004.

[77] Katende E and Jutan A. Nonlinear predictive control. *Proceedings of the American Control Conference*, 6(6):4199–4203, 1995.

[78] Kennedy J. Small worlds and mega-minds: Effects of neighborhood topology on particle swarm performance. *Proceedings of the 1999 IEEE Congress of Evolutionary Computation*, 3:1931–1938, 1999.

[79] Kennedy J. Some issues and practices for the particle swarms. *Proceedings of the 2007 IEEE Swarm Intelligence Symposium*, pages 162–169, 2007.

[80] Kennedy J and Eberhart R. *Swarm intelligence.* Academic Press, 2001.

[81] Kennedy J and Eberhart RC. Particle swarm optimization. *Proceedings of the IEEE International Conference on Neural Networks*, pages 1942–1948, 1995.

[82] Kocijan J, Leith DJ, and West D. Derivative observations used in predictive control. *Proceedings of the 12th IEEE Mediterranean Electrotechnical Conference*, 1:379–382, 2004.

[83] Kong L and Xiao L. A new model predictive control scheme-based load-frequency control. *IEEE International Conference on Control and Automation*, 1:2514–2518, 2007.

[84] Kouvaritakis B, Cannon M, and Rossiter JA. Nonlinear model based predictive control. *International Journal of Control*, 72(10):919–928, 1999.

[85] Kumar A, Malik OP, and Hope GS. Variable-structure-system control applied to agc of an interconnected power system. *IEE Proceedings on Generation, Transmission and Distribution*, 132(1):23–29, 1985.

[86] Kumar A, Malik OP, and Hope GS. Discrete variable-structure controller for load frequency control of multi-area interconnected power system. *IEE Proceedings on Generation, Transmission and Distribution*, 134(2):116–122, 1987.

[87] Lazar M and Pastravanu O. A neural predictive controller for nonlinear systems. *Mathematics and Computers in Simulation*, 60:315–324, 2002.

[88] Lee JH and Morari M. Model predictive control: Past, present and future. *Computers and Chemical Engineering*, 23:667–682, 1999.

[89] Liaw CM. Design of a reduced-order adaptive lfc for an interconnected hydrothermal power system. *International Journal of Control*, 60(6):1051–1063, 1994.

[90] Liu GP, Kadirkamanathan V, and Billings SA. Predictive control for nonlinear systems using neural networks. *International Journal of Control*, 71(6):1119–1132, 1998.

[91] Løvbjerg M, Rasmussen TK, and Krink T. Hybrid particle swarm optimiser with breeding and subpopulations. *Proceedings of the Third Genetic and Evolutionary Computation Conference*, pages 469–476, 2001.

[92] Magni L. On robust tracking with nonlinear model predictive control. *International Journal of Control*, 75(6):399–407, 2002.

[93] Magni L, Bastin G, and Wertz V. Multivariable nonlinear predictive control of cement mills. *IEEE Transactions on Control Systems Technology*, 7(4):502–508, 1999.

[94] Mathur HD and Ghosh S. A comprehensive analysis of intelligent controllers for load frequency control. *IEEE Power India Conference*, 2006.

[95] Matthews GP, DeCarlo RA, Hawley P, and LeFebvre S. Toward a feasible variable structure control design for a synchronous machine connected to an infinite bus. *IEEE Transactions on Automatic Control*, 31(12):1159–1163, 1986.

[96] Mayne D. Model predictive control: the challenge of uncertainty. *IEE Two-Day Workshop on Model Predictive Control: Techniques and Applications*, pages 1–5, 1999.

[97] Mohtadi C, Clarke DW, and Tuff PS. Generalized predictive control. pan 1: The basic algorithm. *Automatica*, 23(2):137–148, 1987.

[98] Mohtadi C, Clarke DW, and Tuff PS. Generalized predictive control. pan 2: Extensions and interpretations. *Automatica*, 23(2):149–160, 1987.

[99] Morton AJ and Price PH. The controlability of steam output, pressure and water level in drum boilers. *Steam at Work: Industrial and Marine Steam Plants*, 11:75–84, 1977.

[100] Naeem W. Nonlinear predictive control using genetic algorithms. *MS Thesis, King Fahd University of Petroleum & Minerals Dhahran, KSA*, 2001.

[101] Pahalawaththa NC, Hope GC, and Malik OP. Multivariable self-tunning power system stabilizer: Simulation and implementation studies. *IEEE Transactions on Energy Conversion*, 6(2):310–319, 1991.

[102] Palazoglu A, Fruzzetti KP, and McDonald KA. Nonlinear model predictive control using hammerstein models. *Journal of Process Control*, 7(1):31–4l, 1997.

[103] Palazoglu A, Norquay SJ, and Romagnoli JA. Model predictive control based on wiener models. *Chemical Engineering Science*, 53(1):75–84, 1998.

[104] Pan CT and Liaw CM. An adaptive controller for power system and load frequency control. *IEEE Transactions on Power Systems*, 4(1):122–128, 1989.

[105] Pandit M, Bergold S, and Pfeiffer B. Unified representation of predictive control algorithms for practical applications. *IEE UKACC International Conference on Control*, pages 787–792, 1998.

[106] Pappa N and Shanmugam J. Neural network based predictor for control of cascaded thermal process. *Proceedings of IEEE International Conference on Intelligent Sensing and Information Processing*, pages 289–294, 2004.

[107] Parsopoulos KE and Vrahatis MN. Particle swarm optimization method in multiobjective problems. *Proceedings of the ACM 2002 Symposium on Applied Computing (SAC 2002)*, pages 603–607, 2002.

[108] Parsopoulos KE and Vrahatis MN. On the computation of all global minimizers through particle swarm optimization. *IEEE Transactions on Evolutionary Computing*, 8(3):211–224, 2004.

[109] Patwardhan SC and Madhavan KP. Nonlinear predictive control of an exothermic cstr using recursivequadratic state space models. *Proceedings of the 4th IEEE Conference on Control Applications*, pages 967–972, 1995.

[110] Peng H, Ozaki T, Toyodat Y, and Haggan-Ozaki V. Nonlinear predictive control based on a global model identified offline. *Proceedings of the 2002 American Control Conference*, 5:4197–4202, 2002.

[111] Pierre DA. A perspective of adaptive control of power system. *Electric Power System and Research*, 2:423–426, 1987.

[112] Prett DM, Garcia CE, and Morari M. Model predictive control: Theory and practice - a survey. *Automatica*, 25(3):335–348, 1989.

[113] Qin SJ and Badgwell TA. An overview of model predictive control applications. Technical Report.

[114] Qin SJ and Badgwell TA. An overview of industrial model predictive control technology. 1997. Available online at: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.52.8909.

[115] Soeterboek R. *Predictive Control - A Unified Approach*. Prentice Hall, 1992.

[116] Rau M and Schroder D. Model predictive control with nonlinear state space models. *International Workshop on Advanced Motion Control*, pages 136–141, 2002.

[117] Rawlings JB. Tutorial overview of model predictive control. *IEEE Control Systems Magazine*, 20(3):38–52, 2000.

[118] Rechenberg I, Zurada JM, Marks II RJ, and Robinson CJ. Evolution strategy, in: Computational intelligence: Imitating life. *Eds., IEEE Press*, 1994. Publication no. 427.

[119] Rerkpreedapong D, Hasanovic A, and Feliachi A. Robust load frequency control using genetic algorithms and linear matrix inequalities. *IEEE Transcations on Power Systems*, 18(2):855–861, 2003.

[120] Richalet J. Industrial applications of model based predictive control. *Automatica*, 29(5):1251–1274, 1993.

[121] Roberts PD. A brief overview of model predictive control. *IEE Two-Day Workshop on Model Predictive Control: Techniques and Applications*, pages 1–4, 1999.

[122] Roset B and Nijmeijer H. Observer-based model predictive control. *International Journal of Control*, 77(17):1452–1462, 2004.

[123] Ross CW. Error adaptive control computer for interconnected power system. *IEEE Transactions on Power Apparatus and Systems*, 85(7):742–749, 1966.

[124] Rossiter JA, Kouvaritakis B, and Gossner JR. Mixed objective constrained stable generalised predictive control. *IEE Proceedings of Control Theory and Applications*, 142(4):286–294, 1995.

[125] Rueda A, Cristea S, de Prada C, and De Keyser R. Non-linear predictive control for a distillation column. *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC '05*, pages 5156–5161, 2005.

[126] Sandou G, Font S, Tebbani S, Hiret A, and Mondon C. Predictive control of a complex district heating network. *Proceedings of the 44th IEEE Conference on Decision and Control*, 2005.

[127] Sbarbaro DG and Jones RW. Multivariable nonlinear control of a paper machine headbox. *Proceedings of the Third IEEE Conference on Control Applications*, 2:769–774, 1994.

[128] Shakil M and Abido MA. Online intelligent control of a boiler-turbine unit using tabu-search algorithm. *International Power Engineering Conference*, pages 1391–1395, 2007.

[129] Shayeghi H, Jalili A, and Shayanfar HA. Multi-stage fuzzy load frequency control using pso. *Journal of Energy Conversion and Management*, 49(1):2570–2580, 2008.

[130] Shayeghi H and Shayanfar HA. Application of ann technique for interconnected power system load frequency control. *International Journal of Energy*, 16(3):246–254, 2003.

[131] Shayeghi H, Shayanfar HA, and Jalili A. Load frequency control strategies: A state-of-the-art survey for the researcher. *Journal of Energy Conversion and Management*, 50(1):344–353, 2009.

[132] Shi XH, Lu YH, Zhou CG, Lee HP, Lin WZ, and Liang YC. Hybrid evolutionary algorithms based on pso and ga. *The 2003 Congress on Evolutionary Computation*, 4:2393–2399, 1996.

[133] Shi YH and Eberhart RC. A modified particle swarm optimizer. *Proceedings of the IEEE International Conference on Evolutionary Computation*, pages 69–73, 1998.

[134] Shi YH and Eberhart RC. Parameter selection in particle swarm optimization. *Evolutionary Programming VII, Springer, Lecture Notes in Computer Science*, 1447:591–600, 1998.

[135] Shi YH and Eberhart RC. Empirical study of particle swarm optimization. *Proceedings of the IEEE International Conference on Evolutionary Computation*, pages 1945–1950, 1999.

[136] Shi YH, Eberhart RC, and Chen YB. Design of evolutionary fuzzy expert system. *Proceedings of the 1997 Artificial Neural Networks in Engineering Conference*, 1997.

[137] Shirai G. Load frequency control using lyapunov's second method: Bang-bang control of speed changer position. *Proceedings of the IEEE*, 67(10):1458–1459, 1979.

[138] Sivaramakrishnan AY, Hariharan MV, and Srisailam MC. Design of a variable structure load frequency controller using pole assignment technique. *International Journal of Control*, 31(5):833–891, 1980.

[139] Sommer R. Control design for multivariable nonlinear time varying systems. *International Journal of Control*, 31(5):833–891, 1980.

[140] Song BJ and Koivo AJ. Nonlinear predictive control with application to manipulator with flexible forearm. *IEEE Transactions on Industrial Electronics*, 46(5):923–932, 1999.

[141] Song Y, Chen ZQ, and Yuan Z. New chaotic pso-based neural network predictive control for nonlinear process. *IEEE Transactions on Neural Networks*, 18(2):595–601, 2007.

[142] Spathopoulos MP and Fragopoulos D. Pendulation control of an offshore crane. *International Journal of Control*, 77(7):654–670, 2004.

[143] Su R. On the linear equivalents of nonlinear systems. *System Control Letters*, 2(1):48–52, 1982.

[144] Tamas L, Nascu L, and de Keyser R. The nepsac nonlinear predictive controller in a real life experiment. *Proceedings of the 11th IEEE International Conference on Intelligent Engineering Systems*, pages 229–234, 2007. Available at: http://hdl.handle.net/1854/LU-383264.

[145] Tan W, Marquez HJ, Chen T, and Liu J. Analysis and control of a nonlinear boiler-turbine unit. *Journal of Process Control*, 15:883891, 2005.

[146] Tao Z, Gang W, Qing L, Weil C, and Feng H. Fast modular multivariable nonlinear model predictive controller. *Proceedings of the 26th Chinese Control Conference*, 2007.

[147] Testud JL, Richalet J, Rault A, and Papon J. Model predictive heuristic control: Applications to industial processes. *Automatica*, 14:413–428, 1978.

[148] Tu KH. A comparative study of multivariable boiler control methods. *Master's Thesis, University of Texas at Austin*, 1992.

[149] Vajk I, Vajta M, and Keviczky L. Adaptive load frequency control of hungarian power system. *Automatica*, 21(2):129–137, 1985.

[150] van den Bergh F and Engelbrecht AP. Cooperative learning in neural networks using particle swarm optimizers. *Annual Research Conference of South African Institute of Computer Scientists and Information Technologists 2000*, 26:84–90, 2001.

[151] van den Bergh F and Engelbrecht AP. Effects of swarm size on cooperative particle swarm optimisers. *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 892–899, 2001.

[152] van den Bergh F and Engelbrecht AP. A cooperative approach to particle swarm optimization. *IEEE Transactions on Evolutionary Computing*, 8(3):225–239, 2004.

[153] Velusami S and Chidambaram IA. Decentralized biased dual mode controllers for load frequency control of interconnected power systems considering gdb and grc non-linearities. *Journal of Energy Conversion and Management*, 48:1691–1702, 2007.

[154] Wang H, Liu Y, Changhe L, and Sanyou Z. A hybrid particle swarm algorithm with cauchy mutation. *Proceedings of the 2007 IEEE Swarm Intelligence Symposium*, pages 356–360, 2007.

[155] Wang Y, Zhou R, and Wen C. Robust load frequency controller design for power systems. *IEE Proceedings Part C*, 140(1):11–16, 1993.

[156] Wei J and Wang Y. A dynamical particle swarm algorithm with dimension mutation. *International Conference on Computational Intelligence and Security*, 1:254–257, 2006.

[157] Wigren T. Recursive indentification based on the nonlinear wiener model. *PhD Thesis, ACTA Universitatis Upsaliensis*, 1990.

[158] Wiley InterScience. Robust and nonlinear control literature survey (no. 1). *International Journal of Robust Nonlinear Control*, 17:1157–1160, 2007. Published Online www.interscience.wiley.com.

[159] Wiley InterScience. Robust and nonlinear control literature survey (no. 2). *International Journal of Robust Nonlinear Control*, 17:1430–1434, 2007. Published Online www.interscience.wiley.com.

[160] Wiley InterScience. Robust and nonlinear control literature survey (no. 3). *International Journal of Robust Nonlinear Control*, 17:1771–1774, 2007. Published Online www.interscience.wiley.com.

[161] Xiao J and Wang X. Nonlinear neural network predictive control for power unit using particle swarm optimization. *Proceedings of the Fifth International Conference on Machine Learning and Cybernetics*, pages 2851–2856, 2006.

[162] Xu J and Xin Z. An extended particle swarm optimizer. *Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium*, 2005.

[163] Xu M and Shaoyuan L. Practical generalized predictive control with decentralized identification approach to hvac systems. *Journal of Energy Conversion and Management*, 48(1):292–299, 2007.

[164] Xu M, Shaoyuan L, Wen-jian C, and Lu L. Effects of a gpc-pid control strategy with hierarchical structure for a cooling coil unit. *Journal of Energy Conversion and Management*, 47(1):132–145, 2006.

[165] Yang TC and Cimen H. Applying structured singular values and a new lqr design to robust decentralized power system load frequency control. *Proceedings of The IEEE International Conference on Industrial Technology*, pages 880–884, 1996.

[166] Yang TC, Cimen H, and Zhu QM. Decentralised load-frequency controller design based on structured singular values. *IEE Proceedings on Generation, Transmission and Distribution*, 145(1):7–14, 1998.

[167] Yesil E, Guzelkaya M, and Eksin I. Self tuning fuzzy pid type load and frequency controller. *Journal of Energy Conversion and Management*, 45:377–390, 2004.

[168] Yim W. Modified nonlinear predictive control of elastic manipulators. *Proceedings of the 1996 IEEE International Conference on Robotics and Automation*, 3:2097–2102, 1996.

[169] Yim W and Singh SN. Predictive end-point trajectory control of elastic manipulators. *Proceedings of the 1995 American Control Conference*, 5:3316–3320, 1995.

[170] Young KD and Kwatny HG. Variable structure servo-mechanism design and applications to overspeed protection control. *Automatica*, 18:385–400, 1982.

[171] Yu YN and Moussa HAM. Optimal stabilization of a multimachine system. *IEEE Transactions on Power Apparatus and Systems*, 91(3):1174–1182, 1972.

[172] Yugeng X, Hao S, and Zhongjun Z. Pso based nonlinear predictive control of single area load frequency control. *Proceedings of the IEEE Region 10 Conference on Computer, Communication, Control and Power Engineering (TENCON)*, pages 428–431, 1993.

[173] Zhang J and Morris J. Nonlinear model predictive control based on multiple local linear models. *Proceedings of the 2001 American Control Conference*, 5:3503–3508, 2001.

[174] Zhao Z, Ikegami Y, Goto S, and Nakamura M. Multi-objective nonlinear predictive control for stec plant. *Proceedings of the 6th World Congress on Intelligent Control and Automation*, 2:6327–6331, 2006.

[175] Zheng A and Allgower F. Towards a practical nonlinear predictive control algorithm withguaranteed stability for large-scale systems. *Proceedings of the 1998 American Control Conference*, 4:2534–2538, 1998.

[176] Zribi M, Al-Rashed M, and Alrifai M. Adaptive decentralized load frequency control of multi-area power systems. *International Journal of Electrical Power & Energy Systems*, 27(8):575–583, 2005.

Vitae

- Muhammad Salman Yousuf.
- Born in Karachi, Pakistan on 18th October, 1983. Pakistani National.
- Education up to high school in Jeddah, Saudi Arabia.
- Received Bachelor of Engineering (B.E.) degree in Electronic Engineer from NED University, Karachi, Pakistan in January 2006.
- Joined the Department of Electrical Engineering at KFUPM as a Research Assistant in September 2006.
- Joined The Procter & Gamble Company at MIC Dammam as Projects Manager in March 2008.
- Completed Master of Science (M.S.) degree in Electrical Engineering from KFUPM in June 2009.
- Email: *salmanyousuf@gmail.com*
- Phone: +966507475781
- Address: P. O. Box 4927 Dammam 31412 KSA